(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 957 452 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
17.11.1999 Bulletin 1999/46

(51) Int Cl.[6]: **G06T 17/00**

(21) Application number: **99302255.7**

(22) Date of filing: **23.03.1999**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **12.05.1998 JP 12800798**

(71) Applicant: **FUJITSU LIMITED**
Kawasaki-shi, Kanagawa 211-8588 (JP)

(72) Inventor: **Sato, Yuichi, c/o Fujitsu Limited**
Kawasaki-shi, Kanagawa, 211-8588 (JP)

(74) Representative: **Fenlon, Christine Lesley et al**
Haseltine Lake & Co.,
Imperial House,
15-19 Kingsway
London WC2B 6UD (GB)

(54) Apparatus and method for checking interference of objects in computer

(57) An interference checking apparatus approximates the shape of an object with an envelope figure, such as a rectangular parallelepiped, a sphere, etc., and checks static interference among many objects at high speed. In this static interference check, an approximate figure is projected into a space of a lower dimension (2), pairs of objects with a possibility of interference are focused on in stages(6), and the number of the pairs is reduced. Then, a precise interference check is performed for the remaining pairs (4).

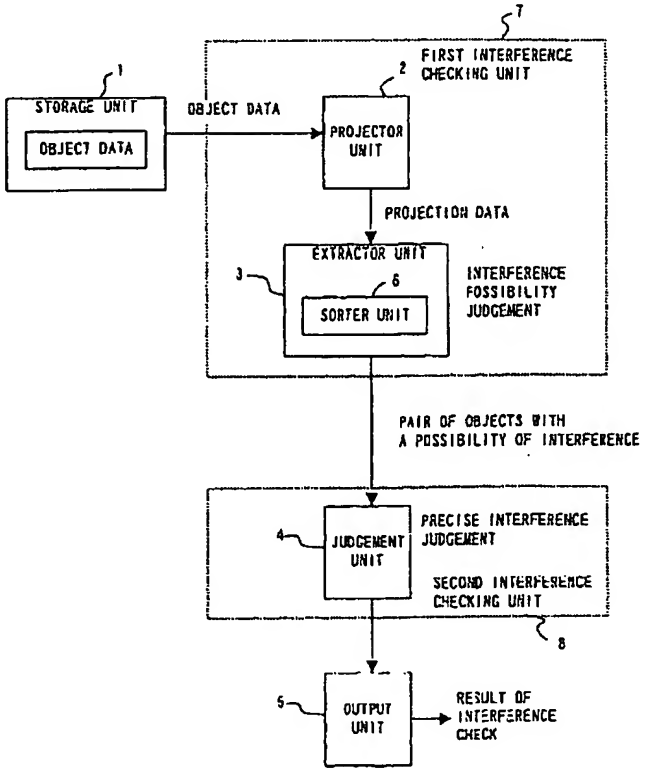

FIG. 1

EP 0 957 452 A2

Description

[0001] The present invention relates to an apparatus for checking the state of interference (overlap) between the computer graphics (CG) models of objects generated in a computer and the method thereof.

[0002] Lately, in mechanical CAD (computer-aided design) systems for simulating the assembled state of products, etc., it has been required to check on the state of interference between the components in an assembly model representing the assembled state and to display the position of interference. Previously-considered checking methods of the state of interference between objects are classified as follows.

(1) A method using an envelope of spheres. The state of interference between objects is judged by enveloping objects with a set of spheres and repeating the check on the interference between the spheres. Technologies corresponding to this method include U.S. Patent Official Gazette 4,578,757 (Stark), U.S. Patent Official Gazette 4,922,430 (Wavish), U.S. Official Gazette 5,056,031 (Nakano et al.), Japanese Patent Laid-open 2-224004 (Kan Nakano et al.), Japanese Patent Laid-open 6-83422 (Kenji Iriguchi et al.), Japanese Patent Laid-open 6-259506 (Miwako Doi et al.) and Japanese Patent Laid-open 7-100733 (Akihiro Goto et al.).

(2) A space is divided into a set of fine grids, and a space map in which codes are allocated to both an area occupied by an object and an unoccupied area, is prepared. Then, the state of interference is judged by retrieving where on the space map other targets are located. A method using an octonal tree belongs to this group. Technologies corresponding to this method include U.S. Patent Official Gazette 5,150,452 (Pollack) and U.S. Patent Official Gazette 5,347,459 (Greenspan et al.) In particular, the method by Greenspan et al. also adopts the method using an envelope of spheres in (1) in combination.

(3) A map consisting of an interfering area and a non-interfering area is drawn on a configuration space, which is well known in a field of robotics, and interference is measured in this space. Technologies corresponding to this method include U.S. Patent Official Gazette, 5,047,916 (Kondo). In Kondo's method, a configuration space is divided into grids, and a map for interference judgement is prepared.

(4) A method based on the calculation of a distance between the nearest points of a convex polyhedron. Technologies corresponding to this method include a method by Gilbert et al. (E.G. Gilbert, D.W. Johnson and S.S. Keerthi, IEEE Journal of Robotics and Automation, Vol.4, No.2, pp.193-203, 1988).

[0003] However, the previously-considered interference checking methods described above have the following problems.

[0004] Out of the previously-considered interference checking methods, methods (1) through (3) target an arbitrary non-convex object, and though they have a wider application range, they have the following problems compared with method (4) which is used for a convex polyhedron.

[0005] Method (1) needs a pretreatment of enveloping an object with spheres, and in the case of a convex polyhedron requires extra calculation compared with the method of Gilbert et al. Since a distance between target objects is not calculated, how much margin is left for interference, etc. cannot be measured.

[0006] In method (2), the pretreatment to prepare a map in which space is divided into grids takes a long time. If the environment changes, the map has to be prepared again, and real time processing for environmental fluctuation cannot be performed. Furthermore, since a space is divided into grids, a position cannot be calculated to an accuracy higher than a unit of one grid.

[0007] In method (3), it takes a long time to prepare a configuration space. As in method (2), when the environment changes, a map has to be prepared again, and real time processing for environmental fluctuation cannot be performed. When a configuration space is divided into grids like Kondo's method, a position cannot be calculated to an accuracy higher than a unit of one grid.

[0008] On the other hand, although method (4) requires no special pretreatment and a distance between target objects can be calculated at high speed, the shape of targets is limited to a convex polyhedron.

[0009] When, along with the recent advent of high-performance computers, etc., three-dimensional simulation is performed, many objects are moved in a three-dimensional virtual space. Therefore, a check on interference between the objects is often needed. It is also needed to verify whether or not an assembly component composed of many discrete components can be realized without any interference between the discrete components.

[0010] However, since methods (1) through (4) are all checking methods for one-to-one interference between objects, a check has to be repeated for every possible combination of many objects when a check on interference among the objects is required.

[0011] Furthermore, since in the case of objects in a recent virtual space, in order to improve reality, an astronomical number of pieces of surface information (polygons) representing an object are used, and a great number of repeated calculations are needed even in the case of a check on one-to-one object interference. When a check on interference among many objects is made using such a calculation method, it takes an enormous calculation time and is not realistic.

[0012] Since, as the number of objects increases, the calculation time needed for an interference check astronomically increases, it is actually impossible to automatically perform a check on interference among many ob-

jects. Under these circumstances, a user manually selects and designates objects to be checked out of many objects, and a check on one-to-one interference between objects is performed for the specific designated objects, if necessary.

[0013] It is desirable to provide an interference checking apparatus for automatically performing an efficient interference check on the CG models of many non-convex objects generated on a computer and the method thereof.

[0014] According to an embodiment of the first aspect of the present invention, the interference checking apparatus comprises a storage unit, a projector unit, an extractor unit, a judgement unit and an output unit, and checks the interference state between a plurality of objects generated on a computer.

[0015] In this embodiment the storage unit stores object data representing a plurality of objects, and the projector unit generates one-dimensional projection data representing each of the plurality of objects using the object data. The extractor unit extracts a pair of objects with a possibility of interference out of the plurality of objects Using the generated projection data, the judgement unit performs the interference check of the extracted pair of objects, and the output unit outputs the result of the interference check.

[0016] According to an embodiment of the second aspect of the present invention, the interference checking apparatus comprises a storage unit, an updating unit, a first judgement unit, a display unit, a calculator unit and a second judgement unit, and checks the interference state between a plurality of objects generated on a computer.

[0017] In this embodiment the storage unit stores both the position information of vertices composing each of the plurality of objects and the parental relation among the plurality of objects, and the updating unit updates the position information of the vertices of a moved object, if necessary, when the object moves. The first judgement unit performs the check of the interference among the plurality of objects at one time based on the updated position information of the vertices, and the display unit graphically displays an interference position where interference is judged to occur. The calculator unit selects one or more objects in the vicinity of the interference position and calculates the shortest distance between the one or more objects and the other objects, and the second judgement unit performs the interference check between the one or more selected objects and the other objects by judging whether or not the obtained shortest distance exceeds a threshold.

[0018] According to an embodiment of the third aspect of the present invention, the interference checking apparatus comprises a first interference checking unit and a second interference checking unit, and checks the interference state of three or more objects generated on a computer.

[0019] In this embodiment, the first interference unit extracts a pair of objects with a possibility of interference out of the three or more objects, and the second interference unit performs the interference check of the extracted pair of objects.

[0020] Reference will now be made, by way of example to the accompanying drawings, in which:

Fig. 1 illustrates the principle of an interference checking apparatus embodying the present invention;
Fig. 2 shows the configuration of the interference checking apparatus of a preferred embodiment;
Fig. 3 shows the configuration of an information processing device ;
Fig. 4 shows recording,media;
Fig. 5 is a flowchart showing the interference checking process;
Fig. 6 shows an assembly model;
Fig. 7 shows a first rectangular parallelepiped;
Fig. 8 shows a second rectangular parallelepiped ;
Fig. 9 shows a coordinate transformation;
Fig. 10 shows a third rectangular parallelepiped;
Fig. 11 shows a fourth rectangular parallelepiped;
Fig. 12 shows displayed interference points;
Fig. 13 shows the listed display of components;
Fig. 14 shows first projection data;
Fig. 15 is a flowchart showing an interference possibility judgement process (No.1);
Fig. 16 is a flowchart showing an interference possibility judgement process (No.2);
Fig. 17 shows a clipping display;
Fig. 18 shows a dynamic determination method of a projection direction;
Fig. 19 shows a first sphere;
Fig. 20 shows a second sphere; and
Fig. 21 shows second projection data.

[0021] Fig. 1 illustrates the principle of an interference checking apparatus embodying the present invention.

[0022] In an embodiment of the present invention, the interference checking apparatus comprises a storage unit 1, a projector unit 2, an extractor unit 3, a judgement unit 4 and an output unit 5, and checks the interference state of a plurality of objects generated on a computer.

[0023] The storage unit 1 stores object data representing a plurality of objects, and the projector unit 2 generates one-dimensional projection data representing each of the plurality of objects using the object data. The extractor unit 3 extracts a pair of objects with a possibility of interference out of the plurality of objects using the generated projection data, the judgement unit 4 performs the interference check of the extracted pair of objects and the output unit 5 outputs the result of the interference check.

[0024] The object data stored by the storage unit 1 includes information on, for example, the shape of an object, the position in an XYZ coordinate system of the object, the parental relation of the objects, etc., and the

projector unit 2 projects each object on each of the X, Y and Z axes and calculates the projection data of a segment. Then, the extractor unit 3 checks the degree of the overlapping of the projection data by comparing a plurality of projection data on each coordinate axis, and selects a pair of objects corresponding to two projection data that overlap each other as a pair with a possibility of interference. Then, the judgement unit 4 performs a more precise interference check on the pair of objects selected like this.

[0025] In this way, by dividing the interference check into two stages of an interference possibility judgement and a precise interference judgement, process targets for the precise interference judgement can be focused on and thereby the efficiency in the interference check among many objects is improved. In the interference possibility judgement, too, the check can be performed at high speed by transforming and simplifying the interference problem of objects in a three-dimensional space, to that in an one-dimensional space.

[0026] Furthermore, the extractor unit 3 includes a sorter unit 6 for sorting the projection data of the plurality of objects, and can also check the overlapping degree of the projection data based on the sorting result of the projection data. In this case, process targets can be reduced in stages by checking the overlapping degree in other directions of the projection data after checking the overlapping on a sorting direction of the projection data, and the efficiency of the interference possibility judgement can be improved.

[0027] In another embodiment of the present invention, the interference checking apparatus comprises a first interference checking unit 7 and a second interference checking unit 8, and checks the interference state of three or more objects generated on a computer.

[0028] The first interference checking unit 7 extracts a pair of objects with a possibility of interference out of the three or more objects, and the second interference checking unit 8 performs the interference check on the extracted pair of objects.

[0029] For example, the calculation speed of the first interference checking unit 7 is higher than that of the second interference checking unit 8, and the first interference checking unit 7 performs the interference checking on all the possible combinations of the three or more objects. For example, the calculation accuracy of the second interference checking unit 8 is higher than that of the first interference checking unit 7, and the second interference checking unit 8 performs the interference check on all the combinations judged by the first interference checking unit 7 to interfere.

[0030] In this way, by dividing the interference check into two stages of a comparatively low-accuracy but high-speed interference possibility judgement and a comparatively low-speed but high-accuracy precise interference judgement, process targets of the precise interference judgement can be focused on, and thereby the efficiency of the interference check of three or more objects can be improved.

[0031] For example, the first interference checking unit 7 includes a projector unit 2 and an extractor unit 3. The projector unit 2 projects object data representing each of the three or more objects a lower-order dimensional space, and the extractor unit 3 extracts a pair of objects with a possibility of interference out of the objects using the projected data. The interference checking unit 8 includes a judgement unit 4.

[0032] For example, the storage unit 1 shown in Fig. 1 corresponds to an object shape/assembly model information storage unit 11 shown in Fig. 2, which is described later, the projector unit 2, extractor unit 3, judgement unit 4, first interference checking unit 7 and second interference checking unit 8 shown in Fig. 1 correspond to a multi-object static interference judgement unit 13, and the output unit 5 shown in Fig. 1 corresponds to an interference position display unit 14.

[0033] In an embodiment of the present invention, whether the CG models of a plurality of objects generated on a computer are separated from or interfere with each other, is judged by the computer, and the nearest point between the objects, interference points among the objects and distances between the objects are calculated at high speed. Then, the processed result is graphically presented to a user on a computer display. It is assumed that target objects for the interference check are arbitrary non-convex objects. The non-convex objects include a non-convex polyhedron.

[0034] An interference check used in an embodiment of the present invention has the following features.

(1) Performs a static interference check among many objects at high speed-by approximating the shape of an object using an envelope figure, such as an rectangular parallelepiped, a sphere, etc.
(2) In the static interference check of (1), after projecting the approximate figure into a lower-dimensional space (for example, on a coordinate axis) and reducing the number of pairs of objects with a possibility of interference by focusing on the pairs in stages, precisely checks interferences of the remaining pairs.
(3) After the static interference check of (1), efficiently identifies an interference position by a clipping process.
(4) After the static interference check of (1), efficiently checks multi-objects to multi-objects interference and efficiently calculates distances between the multi-objects.

[0035] Generally speaking, since in a precise interference judgement using the accurate shape of an object, calculation is repeated for many polygons of an object, the accuracy is high, but it takes an enormous process time. However, if the number of the polygons of the object is reduced by using an approximate figure, the accuracy becomes low, but a high-speed interference

check becomes available. Therefore, first, pairs of objects with a possibility of interference are selected out of many objects by an interference check using an approximate figure, and then a precise interference check is performed using an accurate shape.

[0036] In this way, the efficiency of an interference check among many objects is improved, and the process time is reduced. As a result, the interference among many objects becomes available using a computer within a practical time period, and the process can be automated.

[0037] In the following preferred embodiments, the interference checking apparatus downloads an assembly model prepared by a mechanical CAD system, etc. in a memory, and performs the static interference check of a plurality of objects contained in the assembly model at high speed. Then, the interference checking apparatus displays candidates of interference points on a display screen. At this time, a user moves an object in the vicinity of the displayed candidates of interference points interactively and the interference checking apparatus checks the interference and calculates the distance in real time.

[0038] Fig. 2 shows the configuration of the interference checking apparatus of a preferred embodiment. The interference checking apparatus shown in Fig. 2 comprises an object shape/assembly model information storage unit 11, an object position/posture information updating unit 12, a multi-object static interference judgement unit 13, an interference position display unit 14, a multi-to-multi object distance calculator unit 15 and an inter-object interference judgement unit 16.

[0039] The object shape/assembly model information storage unit 11 stores information on assembly models and the shape of objects. This information includes a set of vertices for representing the shape of objects, a parental relation for representing the assembly relation among objects, the position of objects, the posture of objects, the joint relation among objects, etc.

[0040] The object position/posture information updating unit 12 calculates the amount of changes with the lapse of time of the shape information for both the position and posture of the objects. For example, when an object moves, the amount of change of each of the vertices of the object is calculated. The multi-object static interference judgement unit 13 performs a combinational check for all the interferences among many objects contained in an assembly model at one time, and outputs the interference judgement data.

[0041] The interference position display unit 14 displays interference positions graphically based on the interference judgement data inputted from the multi-object static interference judgement unit 13. The multi-to-multi object distance calculator unit 15 calculates the shortest multi-to-multi (including one-to-multi) object distance of all the combinations of objects displayed by the interference position display unit 14 in real time. The inter-object interference judgement unit 16 judges the interference state among the objects in real time based on the distance information calculated by the multi-to-multi object distance calculator unit 15.

[0042] The interference checking apparatus shown in Fig. 2 is configured using an information processing device (computer) shown in Fig. 3. The information processing device shown in Fig. 3 comprises a CPU (central processing unit) 21, a storage device 22, a RAM (random access memory) 23, a data input device 24, a graphic display 25, an input interface (I/F) 26 and an output interface 27, and they are connected through a bus system.

[0043] The storage device 22 includes a ROM (read only memory), a magnetic disk device, an optical disk device, a magneto-optical disk device, etc., and stores data and programs (execution modules) needed in an interference checking process.

[0044] The storage device 22 stores data, such as operation environment, object shapes, assembly models, the initial positions of objects, execution modules for executing a process corresponding to each unit shown in Fig. 2, execution modules for displaying the movement of objects graphically, etc.

[0045] The CPU 21 performs the coordinate transformation of objects, a static interference check, a nearest vector calculation, an operation to display the calculation result, an operation to display the movement of objects graphically and all other logical operations using the data and execution modules read from the storage device 22. Since a process of updating the position and posture of objects in a space can be considered as a process of transforming the coordinate system of an object to another coordinate system, such a process is called the coordinate transformation of an object here.

[0046] Each of the object position/posture information updating unit 12, a multi-object static interference judgement unit 13, an interference position display unit 14, a multi-to-multi object distance calculator unit 15 and an inter-object interference judgement unit 16 shown in Fig. 2 is mounted as a software component corresponding to a specific program code stored in the storage device 22, and performs its own assigned process by running the program.

[0047] The RAM 23 is a resident memory, and comprises a shape data/assembly data memory 31, a coordinate transformation data memory 32, a static interference data memory 33 and a nearest vector data memory 34.

[0048] The shape data/assembly data memory 31 corresponds to the object shape/assembly model information storage unit 11 shown in Fig. 2, and stores data, such as operation environment, object shape, etc. read from the storage device 22 by the CPU 21. The coordinate transformation data memory 32 stores data on the position and posture of objects calculated by the CPU 21 based on the movement instruction inputted through the data input device 24.

[0049] The static interference data memory 33 stores

the result of a static interference check performed by the CPU 21 using the data of both the shape data/assembly data memory 31 and the coordinate transformation data memory 32. The nearest vector data memory 34 stores the nearest vector data calculated by the CPU 21 based on the result of the static interference check. The data include the nearest points between objects, the distance between objects, the result of interference judgement, etc.

[0050] The data input device 24 includes a keyboard, a pointing device, a touch panel, etc., and inputs information designated by a user, such as operation environment, the position/posture of an object, a movement instruction, etc. to the CPU 21. The graphic display 25 graphically displays each of the data of the memories 32, 33 and 34 in the RAM 23 received through the interface 27. The graphic display can also output inquiries to a user.

[0051] Fig. 4 shows the system configuration using computer-readable recording media for supplying the information processing device shown in Fig. 3 with programs and data. A portable recording medium 41 includes a memory card, a floppy disk, a CD-ROM (compact disk read only memory), an optical disk, a magneto-optical disk, etc., and stores the programs and data described above. The information processing device downloads programs and data from the portable recording medium 41 through a medium driver device 42, and executes the necessary processes using the programs and data.

[0052] The information processing device can also download the programs and data from an external database 44 through a network connecting device 43, and can execute necessary processes using the programs and data. In this case, the network connecting device 43 communicates with external devices through an arbitrary network (line), such as a LAN (local area network), etc. and performs data exchange accompanying the communication.

[0053] Fig. 5 is a flowchart showing the interference checking process by the object position/posture information updating unit 12, a multi-object static interference judgement unit 13 and an interference position display unit 14 shown in Fig. 2.

[0054] First, the object position/posture information updating unit 12 reads object shape data, assembly data, the initial position of each object, etc. being a target for a static interference check, performs the initial settings, and displays them on a screen (step S1). At this time, the object position/posture information updating unit 12 calculates a figure enveloping each component of an assembly model represented by the assembly data, and displays the figure together with the component. When a rectangular parallelepiped is used for this envelope figure, the position of the vertices of the rectangular parallelepiped is calculated.

[0055] For example, in the case of an assembly model shown in Fig. 6, the set of the vertices of a rectangular parallelepiped 53 enveloping a component 51 becomes (a1, a2, a3, a4, a5, a6, a7, a8) as shown in Fig. 7, and the set of the vertices of a rectangular parallelepiped 54 enveloping a component 52 becomes (b1, b2, b3, b4, b5, b6, b7, b8) as shown in Fig. 8.

[0056] Then, the object position/posture information updating unit 12 performs the coordinate transformation (update) of the position and posture of an assembly model according to a user's movement instruction (step S2). At this time, simultaneously the object position/posture information updating unit 12 performs the same coordinate transformation of a figure enveloping each component, and calculates a new figure further enveloping the figure resulted from the transformation.

[0057] For example, when a coordinate transformation shown in Fig. 9 is performed for the assembly model shown in Fig. 6, the original rectangular parallelepipeds 53 and 54 are transformed into those inclined against an XYZ coordinate system as shown in Figs. 10 and 11. For this reason, rectangular parallelepipeds 55 and 56 for enveloping these inclined rectangular parallelepipeds are calculated and displayed.

[0058] To simplify processes in later stages, each edge of these rectangular parallelepipeds is set parallel to one of the X axis, Y axis and Z axis. The set of the vertices of the rectangular parallelepiped 55 shown in Fig. 10 becomes (a1', a2', a3', a4', a5', a6', a7' a8'), and the set of the vertices of the rectangular parallelepiped 56 shown in Fig. 11 becomes (b1', b2', b3', b4', b5', b6', b7', b8').

[0059] Then, the multi-object static interference judgement unit 13 projects the calculated envelope figure on each of the X, Y and Z axes, and calculates the projection data (step S3). The projection data correspond to a segment on each coordinate axis, and indicate the scope of a space in which the envelope figure exists in the XYZ coordinate system. For projection data, for example, both the maximum and minimum values of each of the X, Y and Z coordinate values of the vertices composing the envelope figure, are used.

[0060] For example, in the case of the rectangular parallelepiped 55 shown in Fig. 10, the coordinate value of vertex a1' nearest to the origin of the coordinate system corresponds to the minimum value, and the coordinate value of vertex a5' farthest from the origin corresponds to the maximum value. In the case of the rectangular parallelepiped 56 shown in Fig. 11, the coordinate value of vertex b1' corresponds to the minimum value, and the coordinate value of vertex b5' corresponds to the maximum value.

[0061] Then, the multi-object static interference judgement unit 13 sorts the projection data of all the envelope figures calculated in this way, in an ascending order of minimum values. Then, using the sorted projection data, the multi-object static interference judgement unit 13 judges the interference possibility among the components, and selects pairs of components with a possibility of interference (step S4).

[0062] In this way, by approximating the shape of a process target component with an envelope figure, any component of an arbitrary shape can be handled, and thereby the processes of the component, such as the calculation of projection data, etc. become easy to perform. However, when the shape of an original component is a comparatively simple figure, such as a rectangular parallelepiped, etc., an envelope figure is not necessarily needed to be used, and the interference possibility can be judged using the original shape as it is.

[0063] Then, a precise judgement on whether or not components interfere with each other, is made by calculating the shortest distance between the components of the selected pairs (step S5). Here, the distance calculation is performed using the shape of the component itself instead of an envelope figure of the component. For the one-to-one calculation of the shortest distance and the one-to-one interference check between components, a variety of methods other than the conventional methods described earlier can also be used.

[0064] For example, Japanese Patent Laid-open 7-134735 ("The Nearest Point Retrieval Method and Pre-treatment Method thereof") discloses a method of performing an inner product evaluation using grid points calculated from directed graph type structure data in the retrieval of the nearest point among polygons. Japanese Patent Laid-open 8-77210 ("Interference Check Pretreatment Method") discloses a method of dividing the polygon set of non-convex polyhedrons into a high-speed partial polygon set of convex elements and applying a method of Gilbert et al. to the interference check among polyhedrons.

[0065] Japanese Patent Laid-open 9-27046 ("Interference Check Method") discloses a method of performing an interference check at high speed using the binary tree of a sphere enveloping an object. Japanese Patent Laid-open 9-128601 ("Inter-object Distance Operation Device and Inter-object Distance Operation Program Recording Medium") discloses a method of calculating a distance between convex objects, with a high accuracy using the position vector of a control point for defining the surface patch of two convex objects.

[0066] If the calculated distance between components is smaller than a predetermined threshold $\in$ which is a sufficiently small value, it is considered that the components interfere with each other, while if not, it is considered that the components do not interfere with each other. Then, the multi-object static interference judgement unit 13 lists up all the pairs of interfering components (interfering pairs) (step S6). At this time, the multi-object static interference judgement unit 13 stores the component names of the interfering pairs, the three-dimensional coordinate value of a point where interference occurs (interference point), the distance between the components, etc. The interference point is provided, for example, at either of the nearest points of two components or in an appropriate position of the vicinity.

[0067] Then, the interference position display unit 14 displays the interference point of each of the listed interfering pairs graphically (step S7), and terminates the process. At this time, as shown in Fig. 12, each interference point is represented by a flashing point, and a distance d between components is displayed for each interference point. Component names corresponding to each interference point are listed up as shown in Fig. 13. Here, the name of the component 51 corresponds to component A, and that of the component 52 corresponds to component B. By employing such a display, a user can easily recognize both the precise distance between components at an interference point and the interference point of each component.

[0068] Next, both sorting and interference possibility judgements performed in steps S3 and S4 are described in detail. It is assumed here that a suffix for identifying each component of an assembly model is i, the farthest vertex from the origin out of the vertices of a rectangular parallelepiped enveloping the component is max(i), and the nearest vertex to the origin is min(i). Furthermore, the coordinate value (X, Y, Z) of these vertices are assumed to be expressed as follows.

$$\max(i) = (\max X(i), \max Y(i), \max Z(i)),$$

$$\min(i) = (\min X(i), \min Y(i), \min Z(i))$$

In this case, projection data projX(i), projY(i) and projZ(i) of this rectangular parallelepiped on the X, Y and Z axes, respectively, are expressed as follows.

$$\mathrm{projX}(i) = (\max X(i), \min X(i)),$$

$$\mathrm{projY}(i) = (\max Y(i), \min Y(i)),$$

$$\mathrm{projZ}(i) = (\max Z(i), \min Z(i))$$

[0069] Fig. 14 shows the result obtained when four rectangular parallelepipeds are projected on an XYZ plane and are further projected on both X and Y axes. In Fig. 14, rectangles 61, 62, 63 and 64 indicate the projections on the XY plane of the rectangular parallelepipeds generated after a coordinate transformation, and rectangles 65, 66, 67 and 68 indicate the projections on the XY plane of the rectangular parallelepipeds generated before the coordinate transformation.

[0070] Assuming that the suffixes of the rectangular parallelepipeds corresponding to the rectangles 61, 62, 63 and 64 are i=0, 1, 2 and 3, respectively, projection data on the X and Y axes of the rectangular parallelepiped, projX(i) and projY(i), respectively, are as follows.

Rectangular 61

projX(0) = (maxX(0), minX(0)),

projY(0) = (maxY(0), minY(0))

Rectangular 62

projX(1) = (maxX(1), minX(1)),

projY(1) = (maxY(1), minY(1))

Rectangular 63

projX(2) = (maxX(2), minX(2)),

projY(2) = (maxY(2), minY(2))

Rectangular 64

projX(3) = (maxX(3), minX(3)),

projY(3) = (maxY(3), minY(3))

[0071] Projection data projZ(0) on the Z axis can also be calculated in the same way. If the X axis is selected for a sorting direction and projX(i) are sorted in the ascending order of minX(i), projX(0), projX(1), projX(2) and projX(3) are arranged in that order. Then, according to the order resulted from the sorting, their interference possibilities are judged using the projection data.

[0072] Figs.15 and 16 are flowcharts showing the interference possibility judgement process in the case where the X axis is selected for a sorting direction. First, in the sorting result, the multi-object static interference judgement unit 13 assigns I_k (k = 1, 2,..., N) to the suffix of projection data in the k-th order (step S11 in Fig. 15), and sets k=1 (step S12). N indicates the number of target components. In the case of the projection data shown in Fig. 14, N=4, and I_1=0, I_2=1, I_3=2 and I_4=3.

[0073] Then, the multi-object static interference judgement unit 13 assigns I_j to the suffix of projection data in the order later than I_k, compares maxX(I_k) and minX(I_j), and selects I_j such that maxX(I_k) > minX(I_j) is satisfied. Since minX(I_k) < minX(I_j) is guaranteed by the sorting, such projection data, projX(I_k) and projX(I_j) are overlapped with each other. Therefore, components I_k and I_j are considered to be a pair candidate with a possibility of interference.

[0074] First, j=k+1 is assigned (step S13), and I_j is assumed to be the suffix of projection data projX(I_j) which is compared with projX(I_k). Then, the multi-object static interference judgement unit 13 judges whether or not maxX(I_k) > minX(I_j) is satisfied (step S14).

[0075] In Fig. 14, since when the rectangule 61 of the first order is compared with the rectangule 62 of the second order, maxX(I_1) = maxX(0) > minX(1) = minX(I_2), component 0 represented by the rectangule 61 and component 1 represented by the rectangule 62 are considered to be a pair candidate with a possibility of interference.

[0076] If maxX(I_k) > minX(I_j), the next judgement that is made is whether or not projection data are overlapped on the Y axis. If minY(I_k) < minY(I_j) or maxY(I_j) < maxY (I_k) is satisfied, projection data projY(I_k) and projY(I_j) are considered not to be overlapped, and components I_k and I_j are excepted from pair candidates with a possibility of interference.

[0077] Here, first, it is judged whether or not maxY(I_k) < minY(I_j) is satisfied (step S15), and if not, it is judged whether or not maxY(I_j) < minY(I_k) is satisfied (step S16).

[0078] When maxY(Ij) < minY(I_k) is also not satisfied, it is judged whether or not projection data are overlapped on the Z axis. If maxZ(I_k) < minZ(I_j) or maxZ(I_j) <minZ(I_k) is satisfied, projection data projZ(I_k) and projZ (I_j) are considered not to be overlapped, and components I_k and I_j are excepted from pair candidates with a possibility of interference.

[0079] Here, first, it is judged whether or not maxZ(I_k) < minZ(I_j) is satisfied (step S17 in Fig. 16), and if not, it is judged whether or not maxZ(I_j) < minZ(I_k) is satisfied (step S18). Then, if maxZ(I_j) < minZ(I_k) is also not satisfied, components I_k and I_j are selected as a pair with a possibility of interference (step S19).

[0080] In the case of the rectangles 61 and 62 shown in Fig. 14, maxY(I_1) = maxY(0) > minY(1) = minY(I_2), and maxY(I_2) = maxY(1) > minY(0) =minY(I_1). Therefore, whether or not components 0 and 1 are left to be a pair with a possibility of interference is finally determined by a relation between projection data on the Z axis.

[0081] Then, j and N are compared (step S20). If j < N, 1 is added to j (step S21), and the processes of steps S14 and after are repeated. If j = N, all the projection data of an order later than I_k can be considered to have been checked. Therefore, the process of I_k is terminated, and this time, k and N-1 are compared (step S22). If k < N-1, 1 is added to k (step S23), and the processes of steps S13 and after are repeated. If k = N-1, all the combinations of the rectangular parallelepipeds can be considered to have been checked. Therefore, the process is terminated.

[0082] If in step S14, the judgement result is NO, and if in steps S15, S16, S17 and S18, the judgement result is YES, it is judged that there is no possibility of interference between components I_k and I_j, and the processes of steps S20 and after are executed. In this way, all the

pairs of components with a possibility of interference out of all the components of the assembly model can be outputted without exception.

[0083] When four rectangles are process targets in Fig. 14, by the judgements in step S14, a pair of component 0 represented by rectangle 61 and component 1 represented by rectangle 62, and a pair of component 1 and component 2 represented by rectangle 63 are considered to be pair candidates with a possibility of interference. Then, by the judgement in steps S15 and S16, the former are left to be a pair candidate with a possibility of interference and the latter is excepted from the candidate list.

[0084] Although in this interference possibility judgement process, the judgements in steps S14, S15, S16, S17 and S18 are continued after the suffix I_j is selected, the judgement process in step S14 can also be executed for all I_j together.

[0085] In step S14, since the overlapping of projection data on the X axis being a sorting direction is checked, by performing this judgement for all I_j together, the number of pair candidates with a possibility of interference can be greatly reduced at high speed. Therefore, judgement targets for another direction can be limited, and a high-speed processing is available without a sorting.

[0086] To simplify the interference possibility judgement process, some or all of the judgements in steps S15, S16, S17 and S18 can also be omitted. However, in this case, there is a possibility that since the number of pairs with a possibility of interference cannot be sufficiently reduced, a time needed for a subsequent precise interference checking process may increase.

[0087] When the Y or Z axis is selected for a sorting direction, too, the interference possibility judgement can be performed in the same way. For example, when in Fig. 14, the Y axis is selected for a sorting direction, and projY(i) is sorted in the ascending order of minY(i), projY (2), projY(0), projY(1) and projY(3) are arranged in that order. In this case, the suffix I_k of projection data becomes I_1=2, I_2=0, I_3=1 and I_4=3, and the overlapping of the projection data is checked in the order of Y, X and Z axes.

[0088] Generally speaking, an arbitrary coordinate axis can be selected for a sorting direction. However, according to an assembly model, a specific direction is often desirable.

[0089] For example, when after a plurality of electronic components are arranged on a printed circuit board, two or more of the printed circuit boards are stacked in one direction, the number of pair candidates with a possibility of interference can be efficiently reduced by selecting the stacking direction for a sorting direction. In the same way, when after a plurality of components are assembled into a subassembly, two or more of the subassemblies are stacked in one direction, it is desirable to select the stacking direction for the sorting direction.

[0090] In the static interference check described above, as shown in Fig. 12, many interference points are generated at one time, and the displayed interference points are often hard to see. Some interference points cannot be often seen from outside, since the interference points are inside the assembly model. Under these circumstances, the interference position display unit 14 provides a function to efficiently identify an interference position employing both the shift of a viewpoint and clipping.

[0091] Both the viewpoint shift and clipping are well known as the basic technology of CG, and for example, are also used in OpenGL being one of the APIs (application programming interfaces) of a three-dimensional CG library.

[0092] For example, when in the list display shown in Fig. 13, a user clicks the box of "Component A, P1" using a pointing device, such as a mouse, etc., component A (component 51 in Fig. 12) is highlighted, and interference point P1 is selectively flashed. At this time, the flashing of other interference points is cancelled.

[0093] In the same way, when the box of "Component B, P1" is clicked, component B (component 52 in Fig. 12) is highlighted, and interference point P1 is selectively flashed. Thus, even when many interference points occur simultaneously, an interference point can be clearly recognized by selecting it.

[0094] When the box of "Component A, P1" or "Component B, P1" is double-clicked, the clipping display of interference point PI is performed. In the clipping display, a display target is divided into two parts with respect to a plane (clipping plane) set in the vicinity of the selected interference point, and one of them is not displayed.

[0095] For example, as shown in fig.17, the part in the viewing direction of an assembly model is clipped by a clipping plane 71 which is perpendicular to the Z axis and nearly touches interference point PI, and is not displayed on the screen. Simultaneously with the clipping, the viewpoint is shifted in the direction of the interference point, and the vicinity of the interference point is automatically enlarged and displayed. Thus, even when the interference point is hidden inside an assembly model, the interference point can be clearly observed from the viewing direction.

[0096] After the vicinity of the interference point is enlarged like this, for example, when a user selects a displayed component using a pointing device and moves the component interactively, both the multi-to-multi object distance calculator unit 15 and the inter-object interference judgement unit 16 automatically start an one-to-multi interference checking process between the moved component and the other components.

[0097] The multi-to-multi object distance calculator unit 15 performs a distance calculation in real time, for example, using an efficient algorithm with a meta-tree. If an obtained distance between components is smaller than a threshold, the inter-object interference judgement unit 16 judges that those components interfere

with each other, and if not, it judges that they do not interfere with each other.

[0098] The algorithm of interference checking with a meta-tree is disclosed in the Japanese Patent Laid-open 9-27046 described earlier. According to this algorithm, a plurality of objects to be checked are classified into two groups, and a multi-to-multi interference check between the groups is performed. First, a sphere for enveloping each object is generated, and further a new sphere for enveloping the generated spheres is generated. In this way, larger envelope spheres are generated in order, these are hierarchically arranged, and a meta-tree is configured for each group.

[0099] Then, a relation between the envelope spheres of one group and those of the other group is checked by way of a depth-first retrieval based on the two metatrees, and finally a pair of close spheres in the bottom layer is obtained. Then, a distance is calculated applying the method of Gilbert et al. to a pair of objects corresponding to the obtained pair of spheres.

[0100] When this algorithm is applied to a one-to-multi interference check, for example, each layer of one meta-tree can be represented by one sphere. When the moved component comprises a plurality of components, a meta-tree corresponding to the components is generated, and a multi-to-multi interference check is automatically performed.

[0101] Assuming that a component 51 is picked up and moved in the state shown in Fig. 17, interference checks between the component 51 and the other components are performed. In this case, since there is only a component 52 other than the component 51, an interference check is performed between these two components.

[0102] The multi-object static interference judgement unit 13 can also judge at high speed whether the interference is a real interference or just touching by performing a tolerance analysis for interfering pairs listed up in step S6 of Fig. 15. The algorithm of the tolerance analysis is disclosed in prior Japanese Patent Application 9-247324 ("Disassembly Route Generator Apparatus, Assembly Route Generator Apparatus and Mechanical Design Support System").

[0103] According to this algorithm, the combinations of interfering polygons are detected from pairs of interfering objects, one polygon is contracted, and the polygon is slightly moved inside the object. Then, an interference check is performed again, and if the objects do not interfere with each other, it is considered that these objects just touch each other. When they still interfere with each other even if one of the polygons moves, it is considered that these objects really interfere with each other.

[0104] Although in the coordinate transformations shown in Figs.10 and 11, a new rectangular parallelepiped is generated for a component after coordinate transformation, and projection data are calculated using the new rectangular parallelepiped, an original rectangular parallelepiped can also be used as it is by dynamically determining the projecting direction along with the coordinate transformation. For example, as shown in Fig. 18, a new X'Y'Z' coordinate system can also be set up using a coordinate axis parallel to each edge of rectangular parallelepipeds 53 and 54 enveloping components 51 and 52, respectively, which are moved in an XYZ coordinate system, and the rectangular parallelepipeds 53 and 54 can be projected on the X', Y' and Z' axes.

[0105] Although in the preferred embodiments described above, a rectangular parallelepiped is used for a figure for enveloping a component for the purpose of an interference possibility judgement, the same processes can be executed even when another figure is used. For example, in the assembly model shown in Fig. 6, the components 51 and 52 can also be enveloped with spheres 81 and 82, respectively, as shown in Figs. 19 and 20.

[0106] In this case, both the maximum values max{X}, max{Y} and max{Z} and the minimum values min{X}, min{Y} and min{Z} of the X, Y and Z coordinate values, respectively, of all the vertices composing a component are calculated, and both the center position of a sphere and a radius r are calculated as follows.

$$\max = (\max\{X\}, \max\{Y\}, \max\{Z\}),$$

$$\min = (\min\{X\}, \min\{Y\}, \min\{Z\}),$$

$$\text{center} = (\max + \min)/2.0,$$

$$r = \| \max - \min \| /2.0$$

where $\| \max - \min \|$ indicates a distance between the max and min. When a sphere is used, the radius does not change in the coordinate transformation in step S2 of Fig.5. So, it is sufficient to update only the position coordinate of the center. Therefore, the amount of calculation is only 1/8 of that in the case where a rectangular parallelepiped with eight vertices is updated.

[0107] Fig.21 shows the result obtained when four spheres are projected on an XY plane and further projected on X and Y axes. In Fig.21, circles 91, 92, 93 and 94 indicate the projections on the XY plane of four spheres enveloping components. Assuming that the suffixes of spheres corresponding to the circles 91, 92, 93 and 94 are i=0, 1, 2 and 3, respectively, projection data on the X and Y axes of these spheres, projX(i) and projY(i) become as follows.

Circle 91

projX(0) = (maxX(0), minX(0)) ,

projY(0) = (maxY(0), minY(0))

Circle 92

projX(1) = (maxX(1), minX(1)),

projY(1) = (maxY(1), minY(1))

Circle 93

projX(2) = (maxX(2), minX(2)),

projY(2) = (maxY(2), minY(2))

Circle 94

projX(3) = (maxX(3), minX(3)),

projY(3) = (maxY(3), minY(3))

[0108] Projection data on the Z axis, projZ(i) can be calculated in the same way. When a sphere is used, maxX(i), maxY(i) and maxZ(i) can be easily calculated by adding a radius r to the X, Y and Z coordinate values of the center, respectively. In the same way, minX(i), minY(i) and minZ(i) can be easily calculated by subtracting a radius r from the X, Y and Z coordinate values of the center, respectively.

[0109] By using projection data obtained like this, the interference possibility judgement processes shown in Figs. 15 and 16 can be executed in the same way as in the case of a rectangular parallelepiped. The subsequent precise interference judgement process, the display of interference points, a clipping, the enlarged display of the vicinity of an interference point, an interactive interference check, a tolerance analysis, etc. in the case of a circle are the same as described above.

[0110] Although in the preferred embodiments described above, the interference check of an assembly model in a mechanical CAD system is mainly described, the present invention is also applicable to all the fields using a CG model. For example, the present invention can also be applied to planning and off-line teaching with a robot, or virtual reality systems such as games using a high-speed graphics work station/personal computer, animation, walk-through, etc.

[0111] When a robot is remotely controlled, it is necessary to avoid unnecessary collision/interference with other objects in the vicinity. When the robot is operated by a program, it is necessary to perform a path planning in advance in such a way that such collision/interference may not occur. In such a case, the interference check of an embodiment of the present invention can be used.

[0112] In a virtual reality system, it is necessary to properly control interference with other objects in such a way that a displayed object may not take unnatural actions, such as passing through a wall, etc. In such a case, the interference check of an embodiment of the present invention can be used.

[0113] According to an embodiment of the present invention, the interference state of many objects generated on a computer can be judged at high speed, and the position where those obj'ects interfere with each other can be displayed. The distance between an object and another object in the vicinity of the object can also be calculated by checking the interference state in real time while selecting and moving one of interfering objects.

[0114] It will be appreciated that computer programs can be stored on a variety of media, such as floppy disks, hard disks, CD ROMs, etc., or downloaded into a computer for example from the Internet. It is intended that the attached claims should encompass computer programs in all these, and similar, forms.

Claims

1. An interference checking apparatus for checking an interference state of a plurality of objects generated on a computer, comprising:

   storing means (1, 11) for storing object data representing the plurality of objects;
   projecting means (2, 13) for generating one-dimensional projection data representing each of the plurality of objects using the object data;
   extracting means (3, 13) for extracting a pair of objects with a possibility of interference from the plurality of objects using the projection data;
   judging means (4, 13) for checking an interference between the extracted pair of objects; and
   outputting means (5, 14) for outputting a result of an interference check..

2. The interference checking apparatus according to claim 1, wherein said extracting means includes sorting means (6) for sorting the projection data of the plurality of objects, and extracts the pair of objects with a possibility of interference by checking an overlapping state of projection data based on a sorting result of the projection data.

3. The interference checking apparatus according to claim 2, wherein

said projecting means projects the plurality of objects on at least two coordinate axes and calculates the projection data; and
said extracting means, after checking the overlapping state of the projection data in a sorting direction, checks the overlapping state of the projection data in another direction.

4. The interference checking apparatus according to any preceding claim, wherein said projecting means includes approximating means for generating an envelope figure for enveloping each of the plurality of objects, and generates projection data corresponding to the envelope figure.

5. The interference checking apparatus according to claim 4, wherein said approximating means uses one of a rectangular parallelepiped and a sphere for the envelope figure.

6. The interference checking apparatus according to claim 4 or 5, further comprising

transforming means (12) for performing a coordinate transformation of the envelope figure, wherein said projecting means generates the projection data from envelope figure after the coordinate transformation.

7. The interference checking apparatus according to any preceding claim, wherein said judging means includes distance calculating means for calculating a distance between two objects in the pair of objects, and judges whether or not the two objects interfere with each other by performing a threshold judgement for the obtained distance.

8. The interference checking apparatus according to any preceding claim, further comprising:

selecting means (24) for, when said outputting means displays a plurality of pairs of interfering objects, selecting one of the pairs of interfereing objects; and
displaying means (14, 25) for dividing a display target with a plane in a vicinity of an interference position of the selected pair, into two parts, allowing one of the two parts not to be displayed and displaying an enlarged vicinity of the interference position.

9. The interference checking apparatus according to any preceding claim, further comprising:

selecting means (24) for, when said outputting means displays a pair of interfering objects, selecting one of the interfering objects; and
interactive interference checking means (15, 16) for interactively checking interference while moving the selected object.

10. An interference checking apparatus for checking an interference state of a plurality of objects generated on a computer, comprising:

storing means (11) for storing both position information of vertices composing each of the plurality of objects and parental relation between an object and an other object;
updating means (12) for, when an object moves, updating the position information of the vertices of the moved object, in case of necessity;
first judging means (13) for performing a combinational interference check among the plurality of objects at one time based on the updated position information on the vertices;
displaying means (14) for graphically displaying an interference position where interference is judged to occur;
calculating means (15) for selecting at least one object in a vicinity of the interference position and calculating the shortest distance between the at least one object and the other objects; and
second judging means (16) for checking interference between the at least one object and the other objects by performing a threshold judgement for the obtained distance.

11. An interference checking apparatus for checking an interference state of at least three objects generated on a computer, comprising:

first interference checking means (7, 13) for extracting a pair of objects with a possibility of interference from the at least three objects; and
second interference checking means (8, 13) for checking an interference between the extracted pair of objects.

12. The interference checking apparatus according to claim 11, wherein

said first interference checking means is faster in operation speed than said second interference checking means and performs an interference check for all combinations among the at least three objects, and
said second interference means is higher in operation accuracy than said first interference checking means and performs an interference check for all combinations judged to interfere with each other by said first interference means.

13. The interference checking apparatus according to claim 11 or 12,wherein

said first interference checking means includes:

projecting means (2) for projecting object data representing each of the at least three objects into a lower dimensiqnal space; and
extracting means (3) for extracting the pair of objects with a possibility of interference from the at least three objects using projection data, and

said second interference checking means includes

judging means (4) for checking the interference between the extracted pair of objects.

14. The interference checking apparatus according to claim 13, further comprising:

storing means (1, 11) for storing the object data projected by said projecting means; and
outputting means (5, 14) for outputting a result of an interference check by said judging means.

15. A computer-readable recording medium (22) on which is recorded a program for causing a computer to generate a plurality of object models and to check an interference state of the plurality of object models, said program comprising the steps of:

generating one-dimensional projection data representing each of the plurality of objects using object data representing the plurality of objects;
extracting a pair of objects with a possibility of interference from the plurality of objects using the projection data; and
checking interference between the extracted pair of objects.

16. A computer-readable recording medium (22) on which is recorded a program for causing a computer to generate at least three object models and to check an interference state of the plurality of object models, said program comprising the steps of:

extracting a pair of objects with a possibility of interference from the at least three objects; and
checking an interference between the extracted pair of objects.

17. An interference checking method of checking an interference state of a plurality of objects generated on a computer, comprising the steps of:

generating one-dimensional projection data representing each of the plurality of objects using object data representing the plurality of objects;
extracting a pair of objects with a possibility of interference from the plurality of objects using the projection data; and
checking an interference between the extracted pair of objects.

18. An interference checking method of checking an interference state of at least three objects generated on a computer, comprising the steps of:

extracting a pair of objects with a possibility of interference from the at least three objects; and
checking an interference between the extracted pair of objects.

19. A computer program which, when running on a computer, is operable to carry out the steps of a method as claimed in claim 17 or 18.

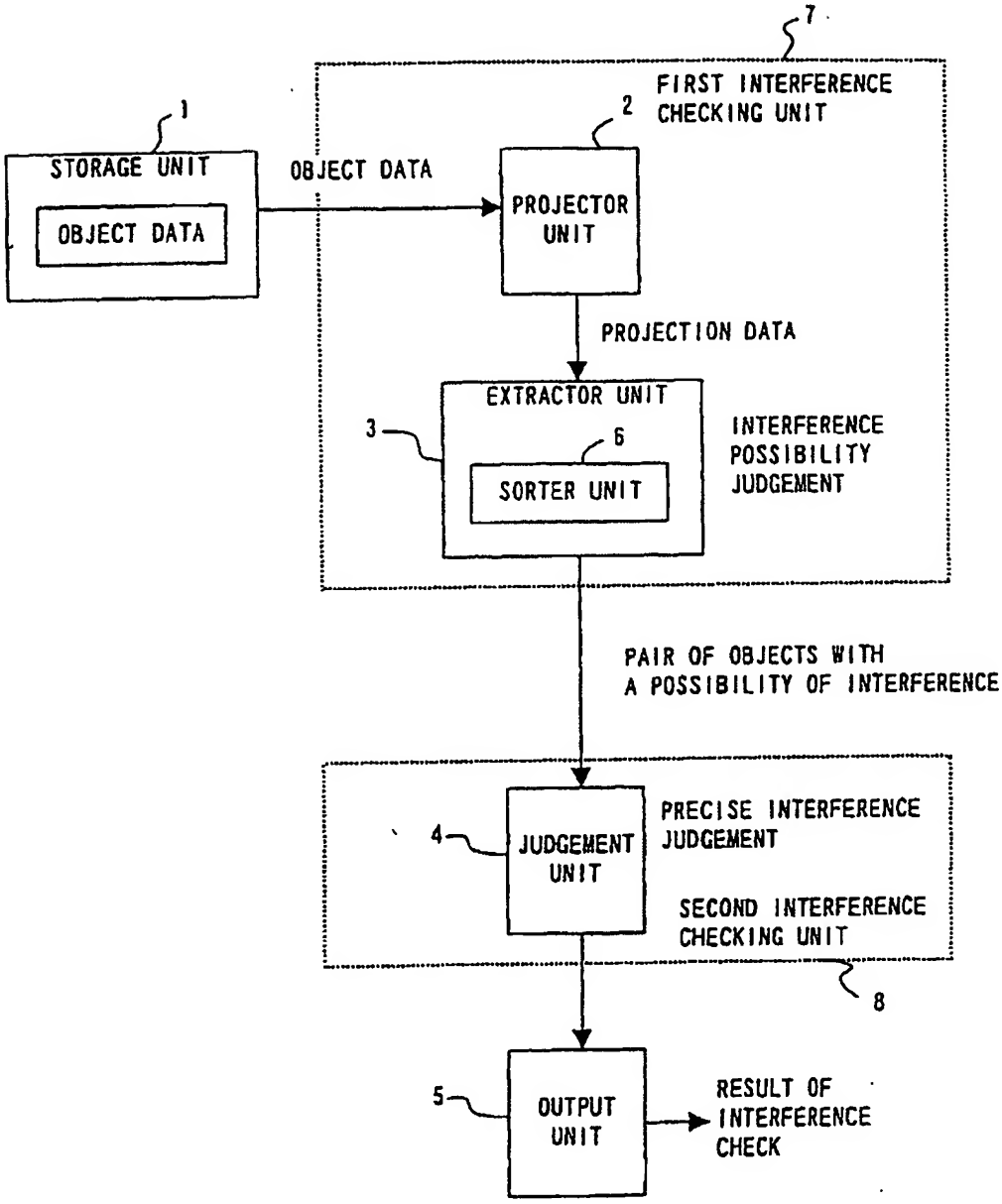
7
FIRST INTERFERENCE CHECKING UNIT
1
STORAGE UNIT
OBJECT DATA
OBJECT DATA
2
PROJECTOR UNIT
PROJECTION DATA
EXTRACTOR UNIT
3
6
SORTER UNIT
INTERFERENCE POSSIBILITY JUDGEMENT
PAIR OF OBJECTS WITH A POSSIBILITY OF INTERFERENCE
4
JUDGEMENT UNIT
PRECISE INTERFERENCE JUDGEMENT
SECOND INTERFERENCE CHECKING UNIT
8
5
OUTPUT UNIT
RESULT OF INTERFERENCE CHECK

FIG. 1

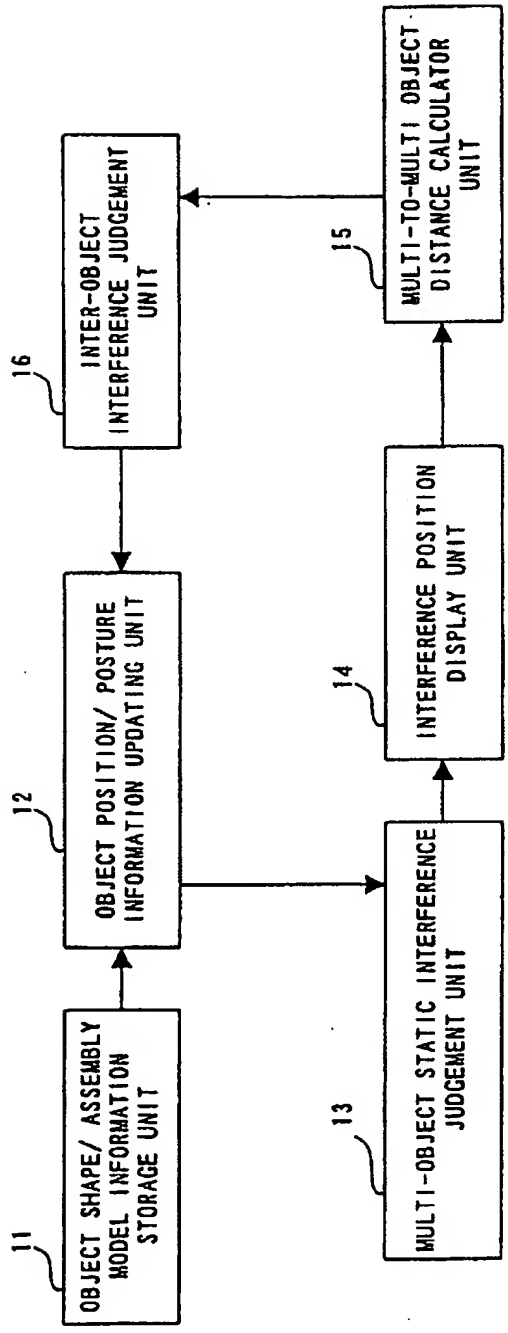
OBJECT SHAPE/ ASSEMBLY MODEL INFORMATION STORAGE UNIT
11
OBJECT POSITION/ POSTURE INFORMATION UPDATING UNIT
12
MULTI-OBJECT STATIC INTERFERENCE JUDGEMENT UNIT
13
INTERFERENCE POSITION DISPLAY UNIT
14
MULTI-TO-MULTI OBJECT DISTANCE CALCULATOR UNIT
15
INTER-OBJECT INTERFERENCE JUDGEMENT UNIT
16

FIG. 2

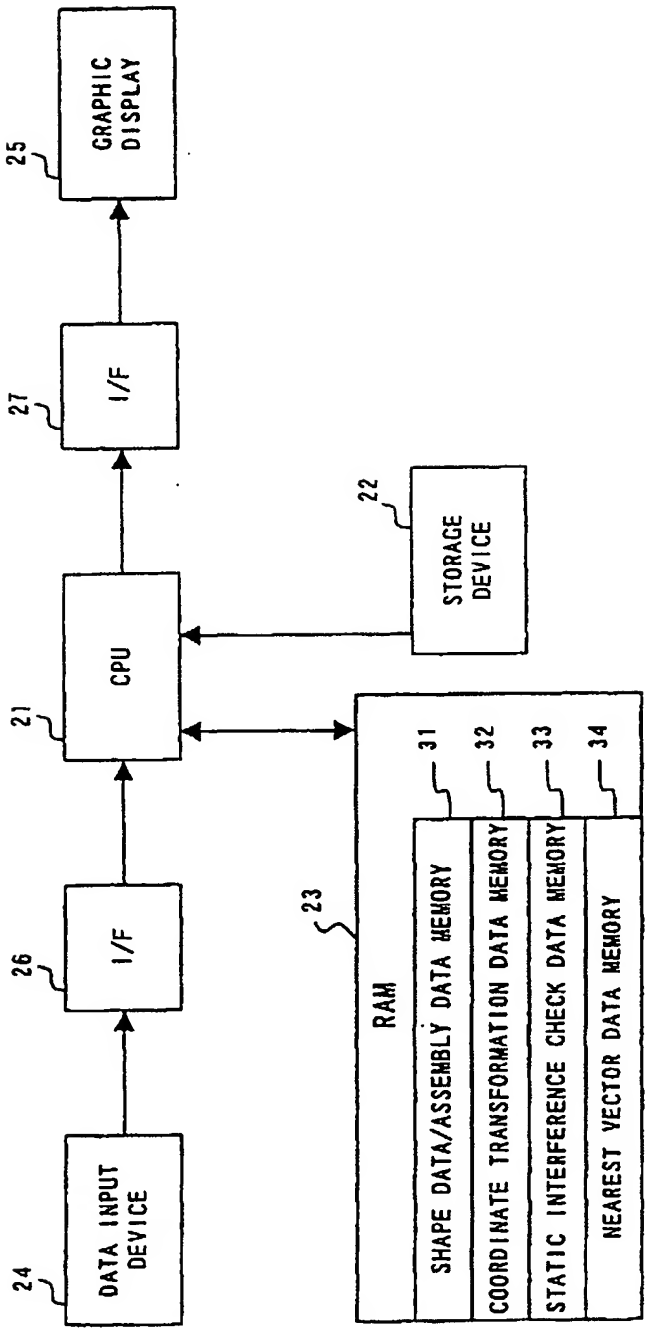
GRAPHIC DISPLAY
25
I/F
27
STORAGE DEVICE
22
CPU
21
I/F
26
RAM
23
SHAPE DATA/ASSEMBLY DATA MEMORY
31
COORDINATE TRANSFORMATION DATA MEMORY
32
STATIC INTERFERENCE CHECK DATA MEMORY
33
NEAREST VECTOR DATA MEMORY
34
DATA INPUT DEVICE
24

FIG. 3

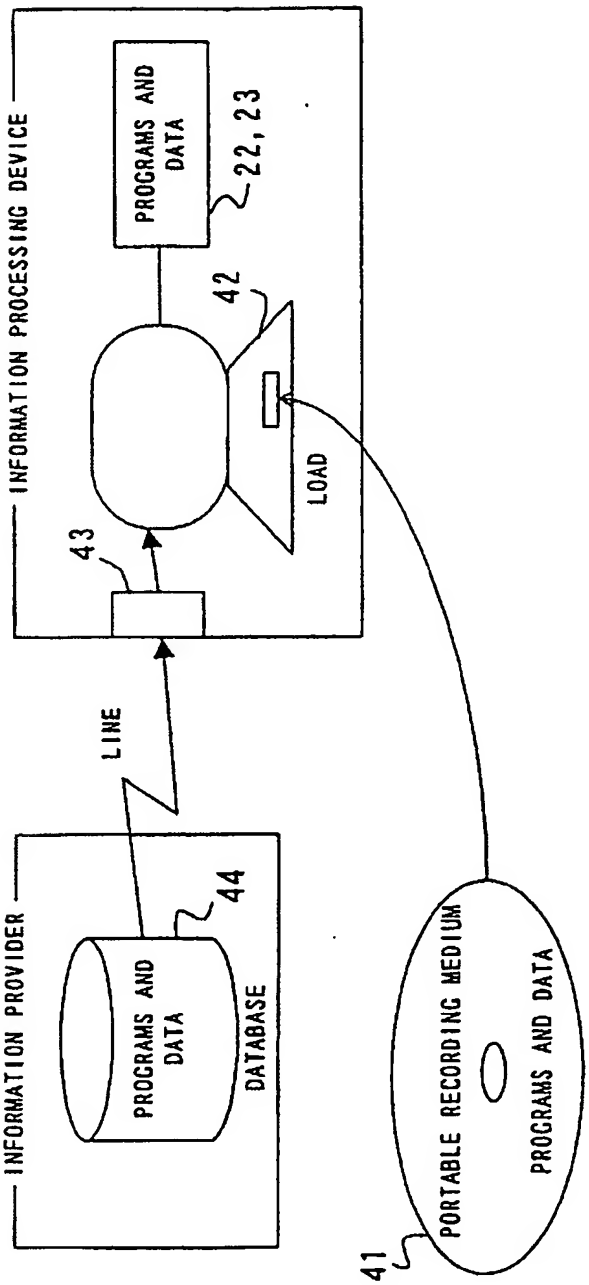
INFORMATION PROCESSING DEVICE
PROGRAMS AND DATA
22, 23
42
LOAD
43
LINE
INFORMATION PROVIDER
44
PROGRAMS AND DATA
DATABASE
PORTABLE RECORDING MEDIUM
PROGRAMS AND DATA
41

FIG. 4

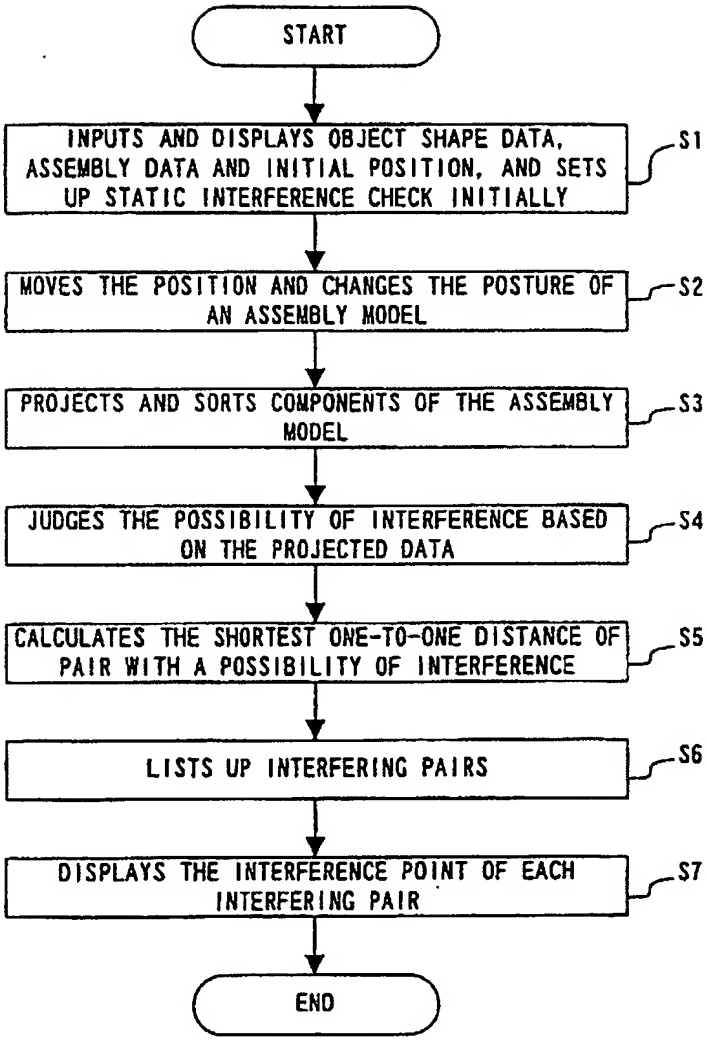
START
INPUTS AND DISPLAYS OBJECT SHAPE DATA, ASSEMBLY DATA AND INITIAL POSITION, AND SETS UP STATIC INTERFERENCE CHECK INITIALLY
S1
MOVES THE POSITION AND CHANGES THE POSTURE OF AN ASSEMBLY MODEL
S2
PROJECTS AND SORTS COMPONENTS OF THE ASSEMBLY MODEL
S3
JUDGES THE POSSIBILITY OF INTERFERENCE BASED ON THE PROJECTED DATA
S4
CALCULATES THE SHORTEST ONE-TO-ONE DISTANCE OF PAIR WITH A POSSIBILITY OF INTERFERENCE
S5
LISTS UP INTERFERING PAIRS
S6
DISPLAYS THE INTERFERENCE POINT OF EACH INTERFERING PAIR
S7
END

FIG. 5

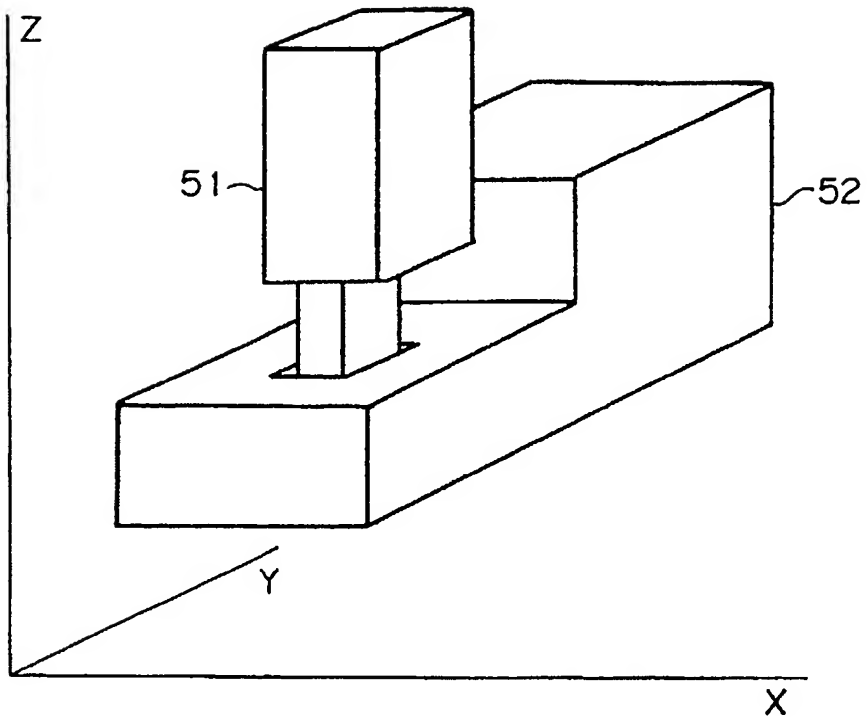

FIG. 6

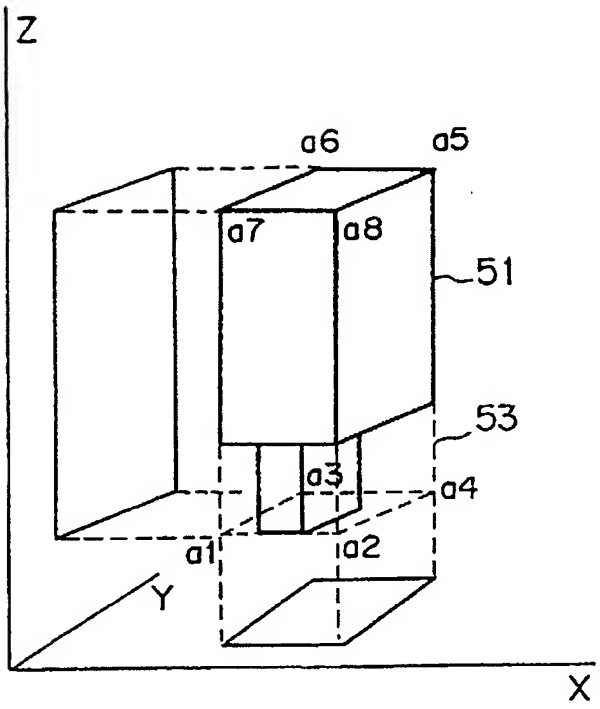

FIG. 7

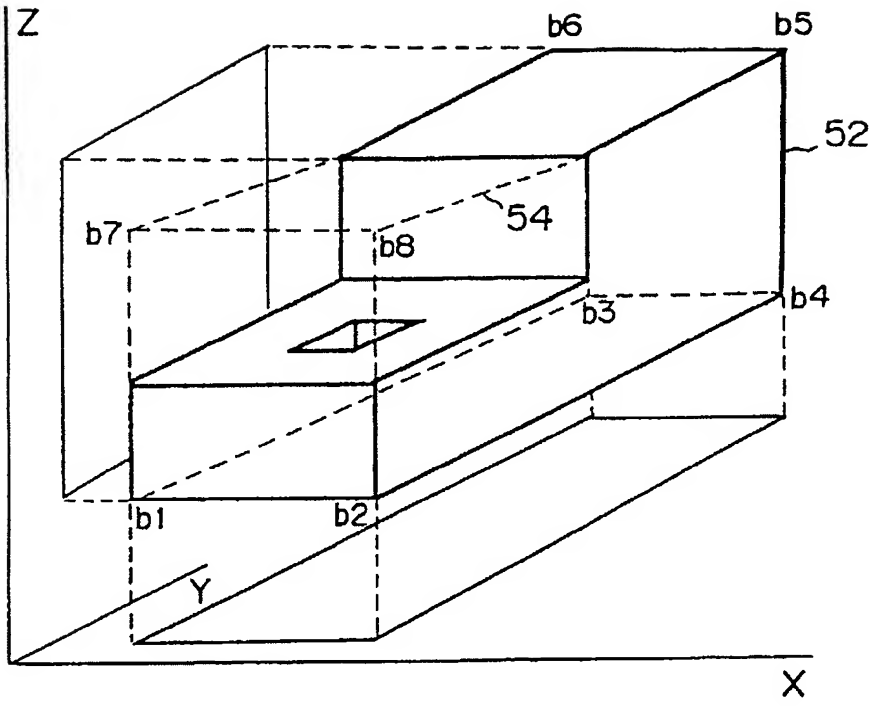

FIG. 8

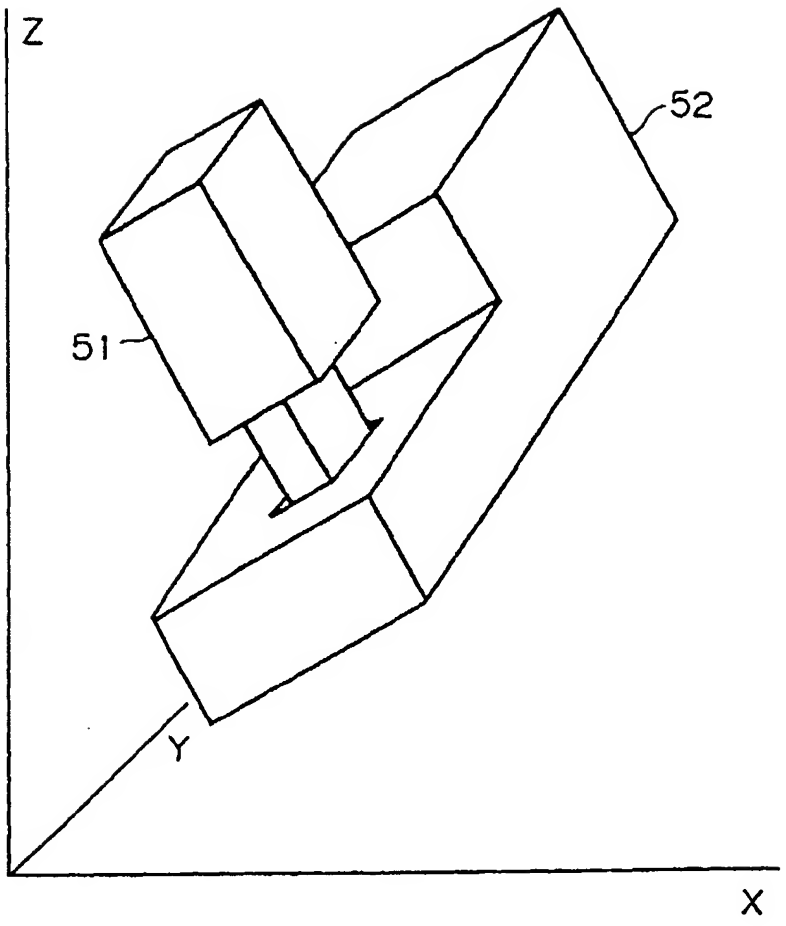

FIG. 9

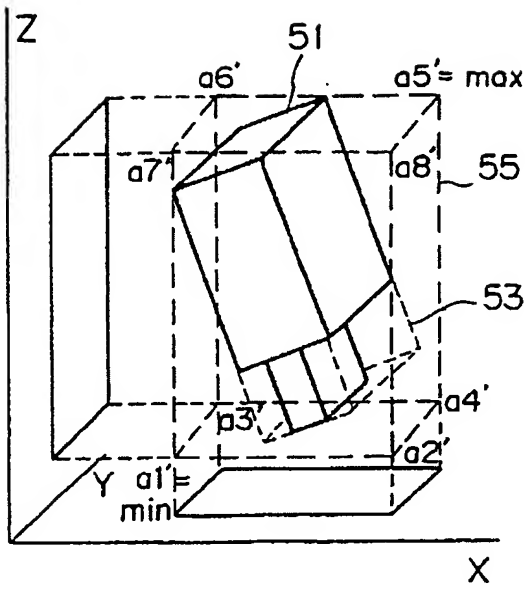

FIG. 10

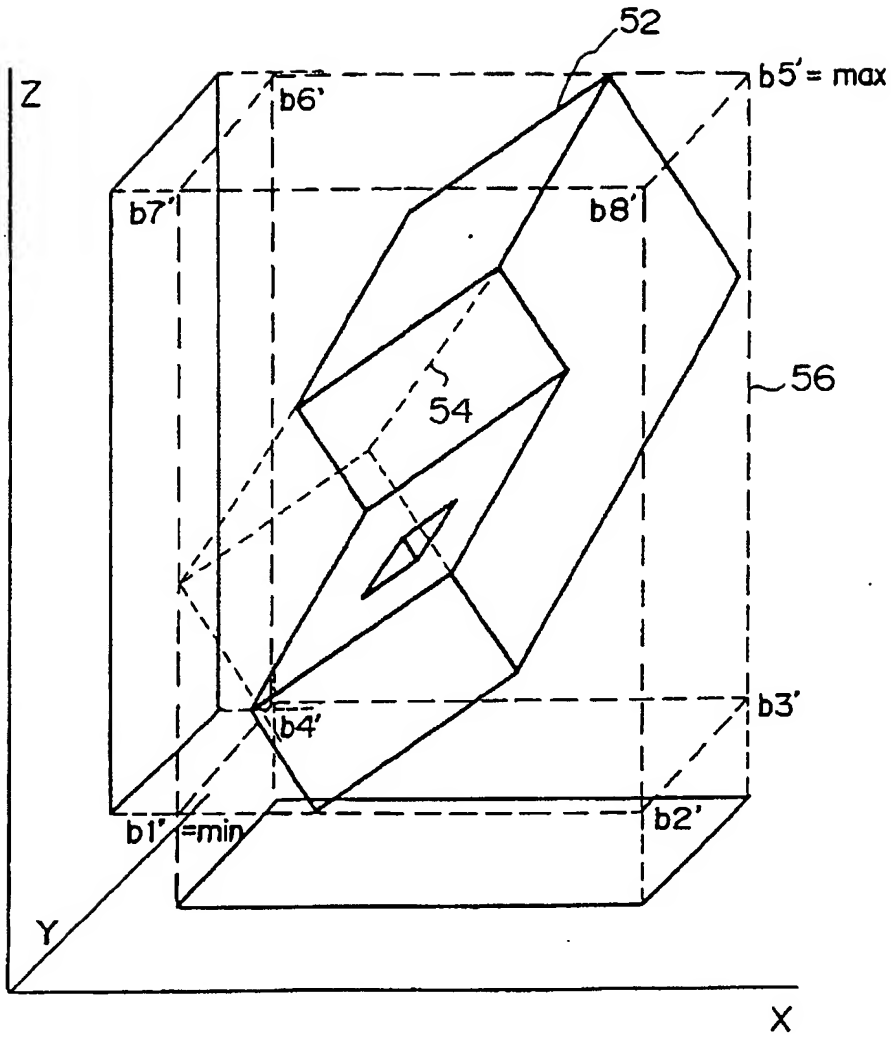

FIG. 11

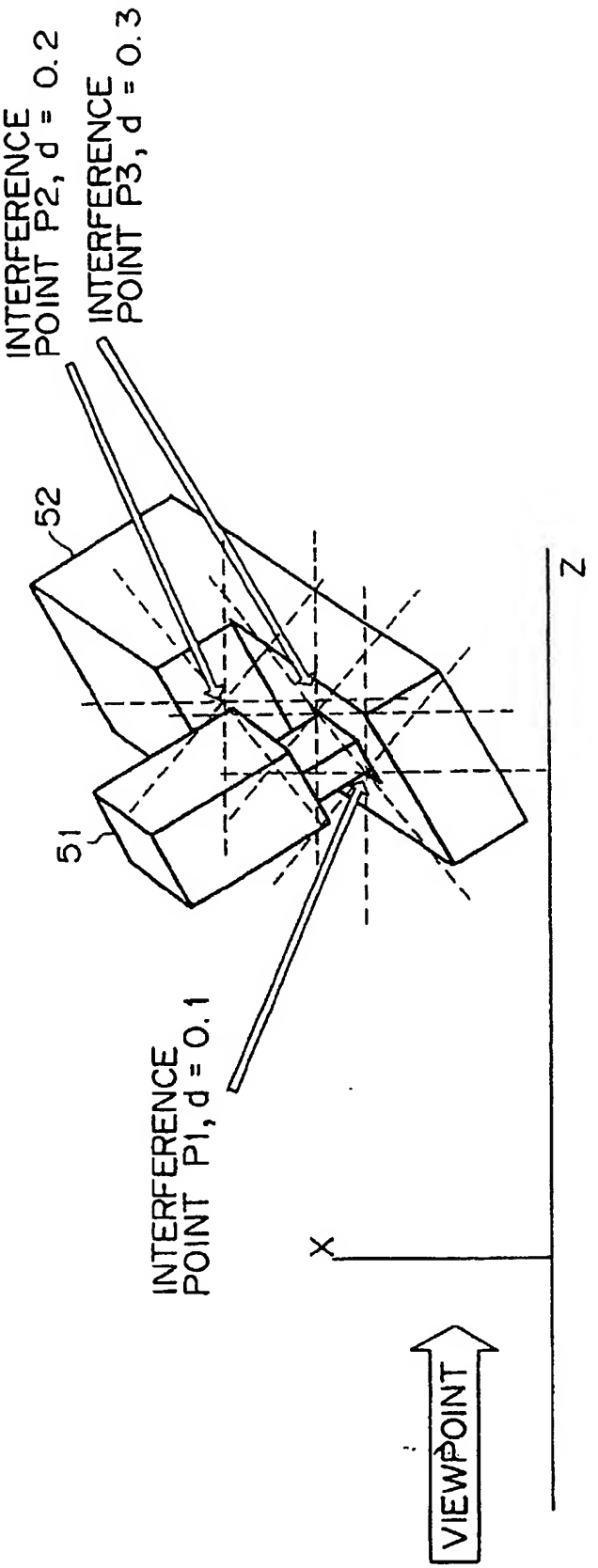
INTERFERENCE POINT P2, d = 0.2
INTERFERENCE POINT P3, d = 0.3
52
51
INTERFERENCE POINT P1, d = 0.1
Z
X
VIEWPOINT

FIG. 12

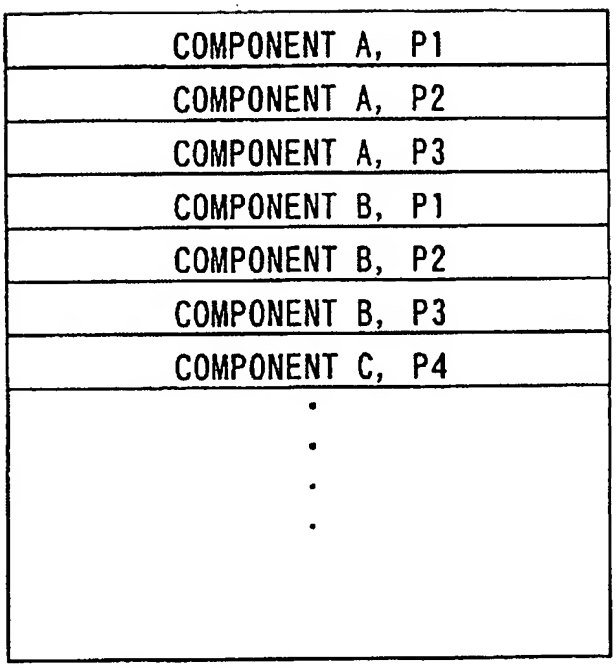
COMPONENT A, P1
COMPONENT A, P2
COMPONENT A, P3
COMPONENT B, P1
COMPONENT B, P2
COMPONENT B, P3
COMPONENT C, P4

FIG. 13

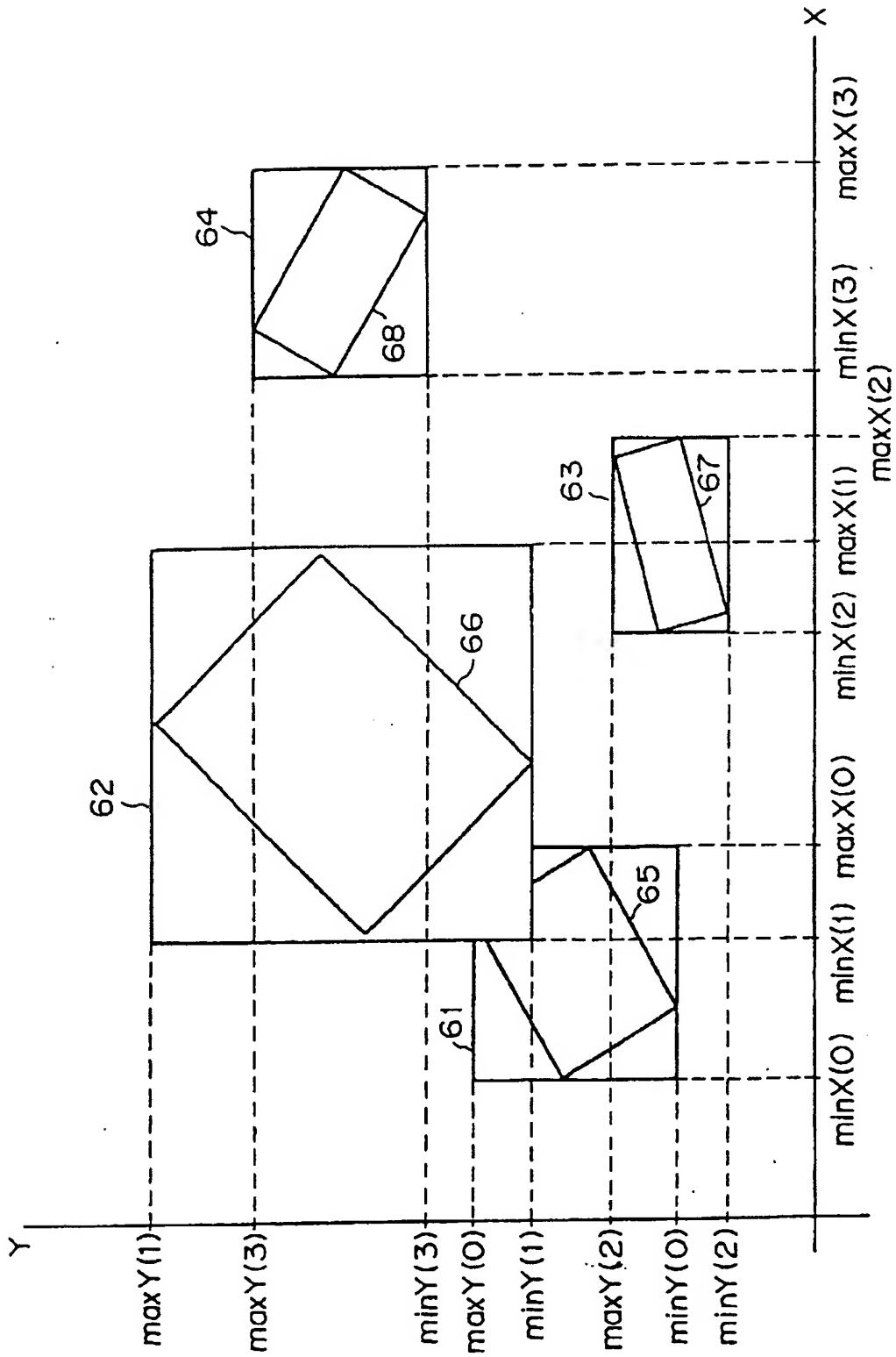
X
Y
maxX(3)
minX(3)
maxX(2)
minX(2)
maxX(1)
minX(1)
maxX(0)
minX(0)
maxY(1)
maxY(3)
minY(3)
maxY(0)
minY(1)
maxY(2)
minY(0)
minY(2)
62
64
68
66
63
67
65
61

FIG. 14

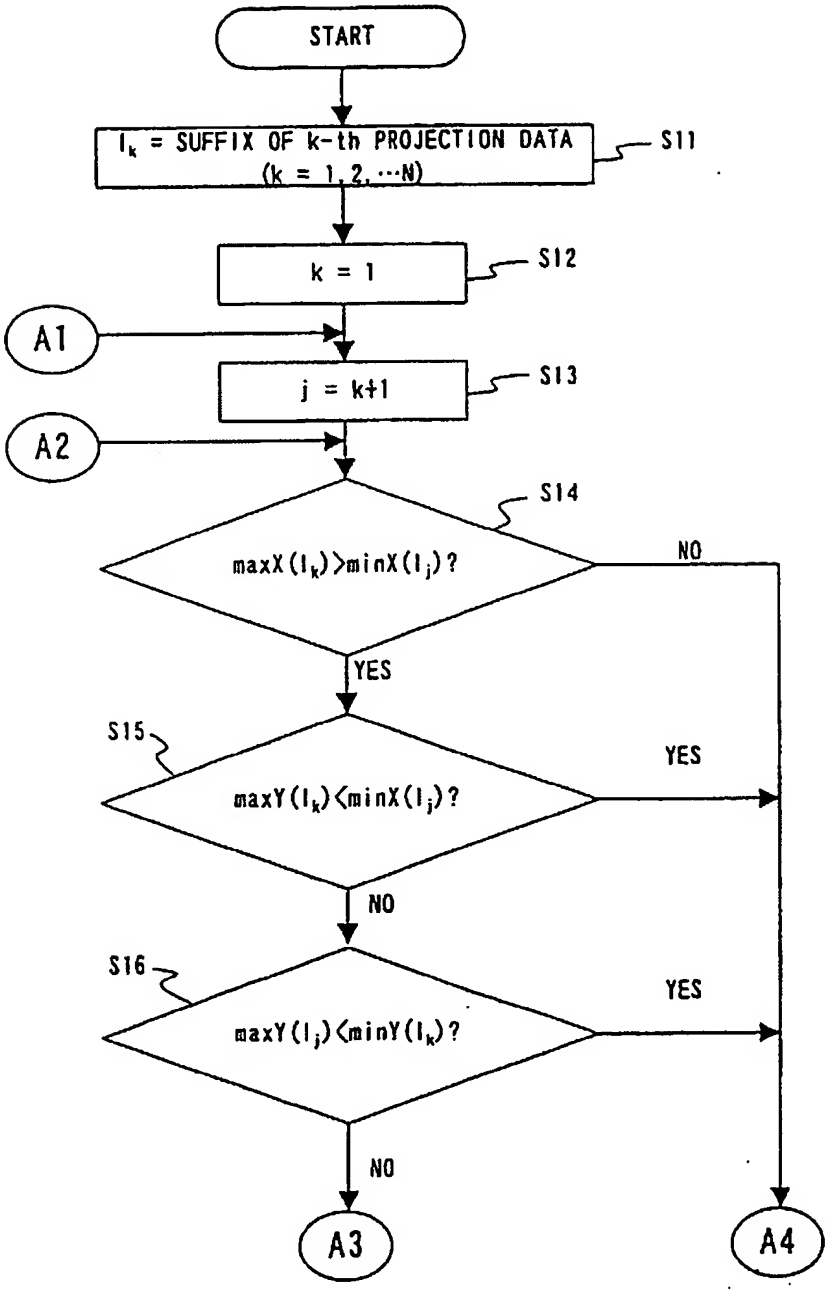
START
I_k = SUFFIX OF k-th PROJECTION DATA (k = 1, 2, ···N)
S11
k = 1
S12
A1
j = k+1
S13
A2
S14
maxX(I_k)>minX(I_j)?
NO
YES
S15
maxY(I_k)<minX(I_j)?
YES
NO
S16
maxY(I_j)<minY(I_k)?
YES
NO
A3
A4

FIG. 15

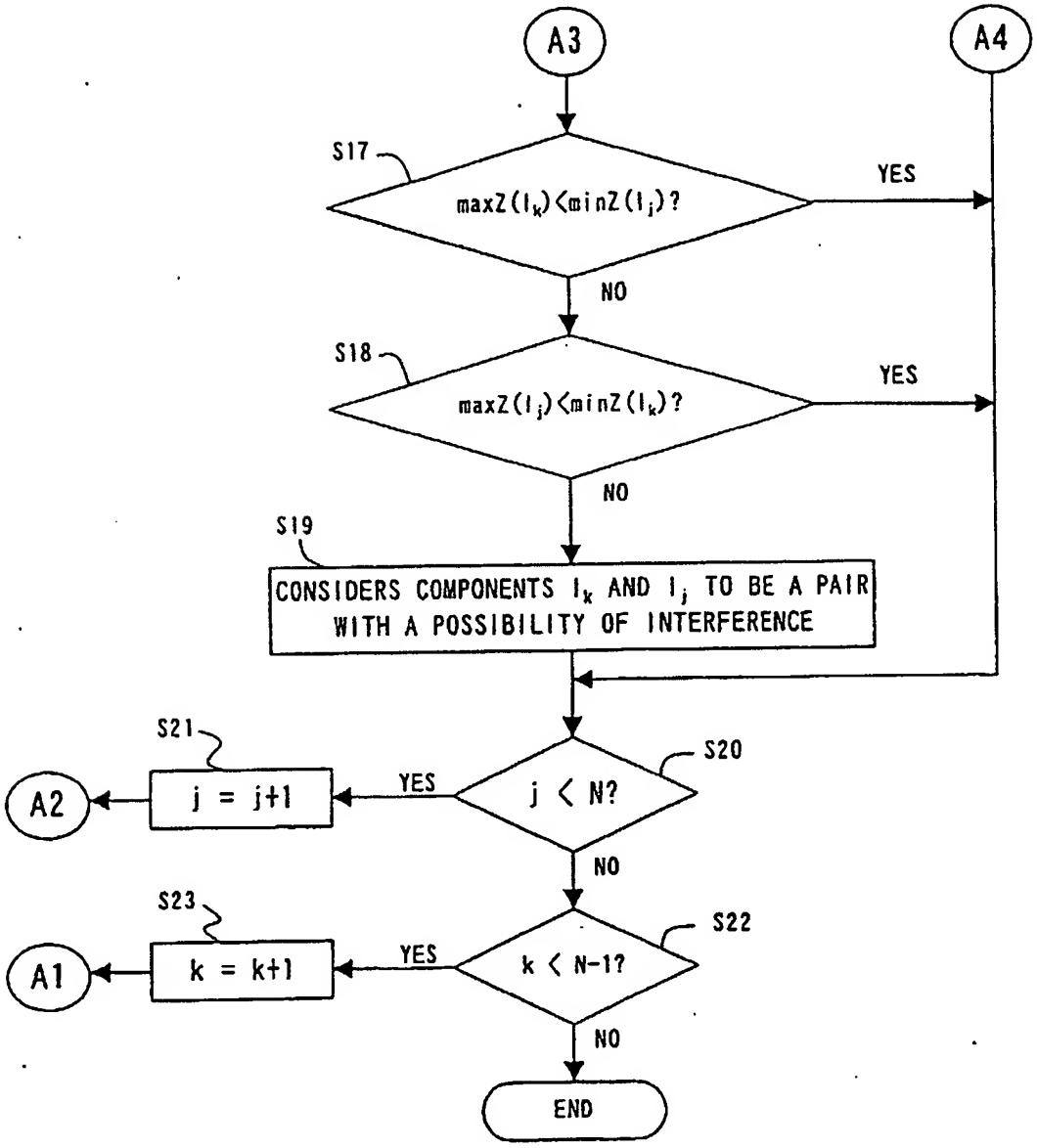
A3
A4
S17
maxZ(I_k)<minZ(I_j)?
YES
NO
S18
maxZ(I_j)<minZ(I_k)?
YES
NO
S19
CONSIDERS COMPONENTS I_k AND I_j TO BE A PAIR WITH A POSSIBILITY OF INTERFERENCE
S21
S20
A2
j = j+1
YES
j < N?
NO
S23
S22
A1
k = k+1
YES
k < N-1?
NO
END

FIG. 16

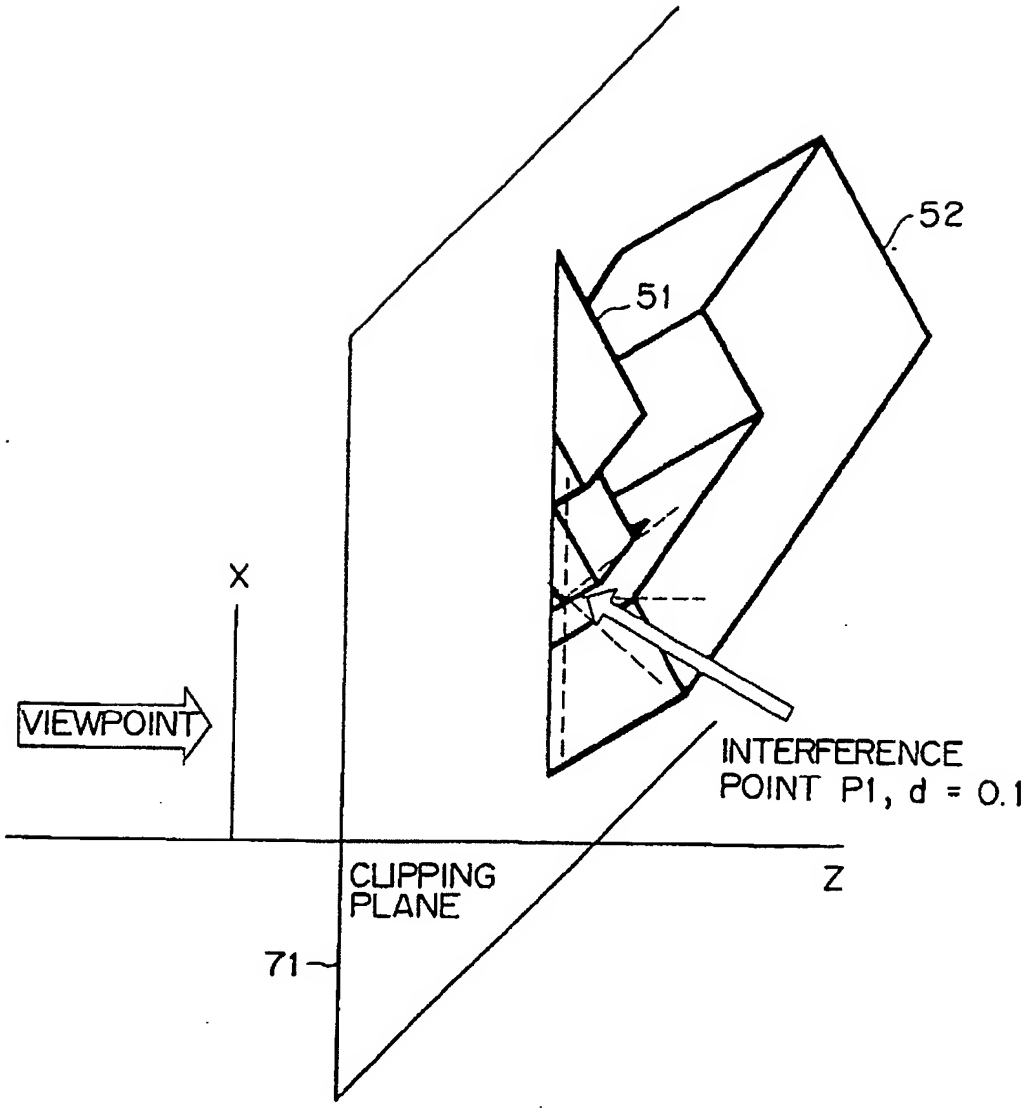
52
51
X
VIEWPOINT
INTERFERENCE
POINT P1, d = 0.1
CLIPPING
PLANE
Z
71

FIG. 17

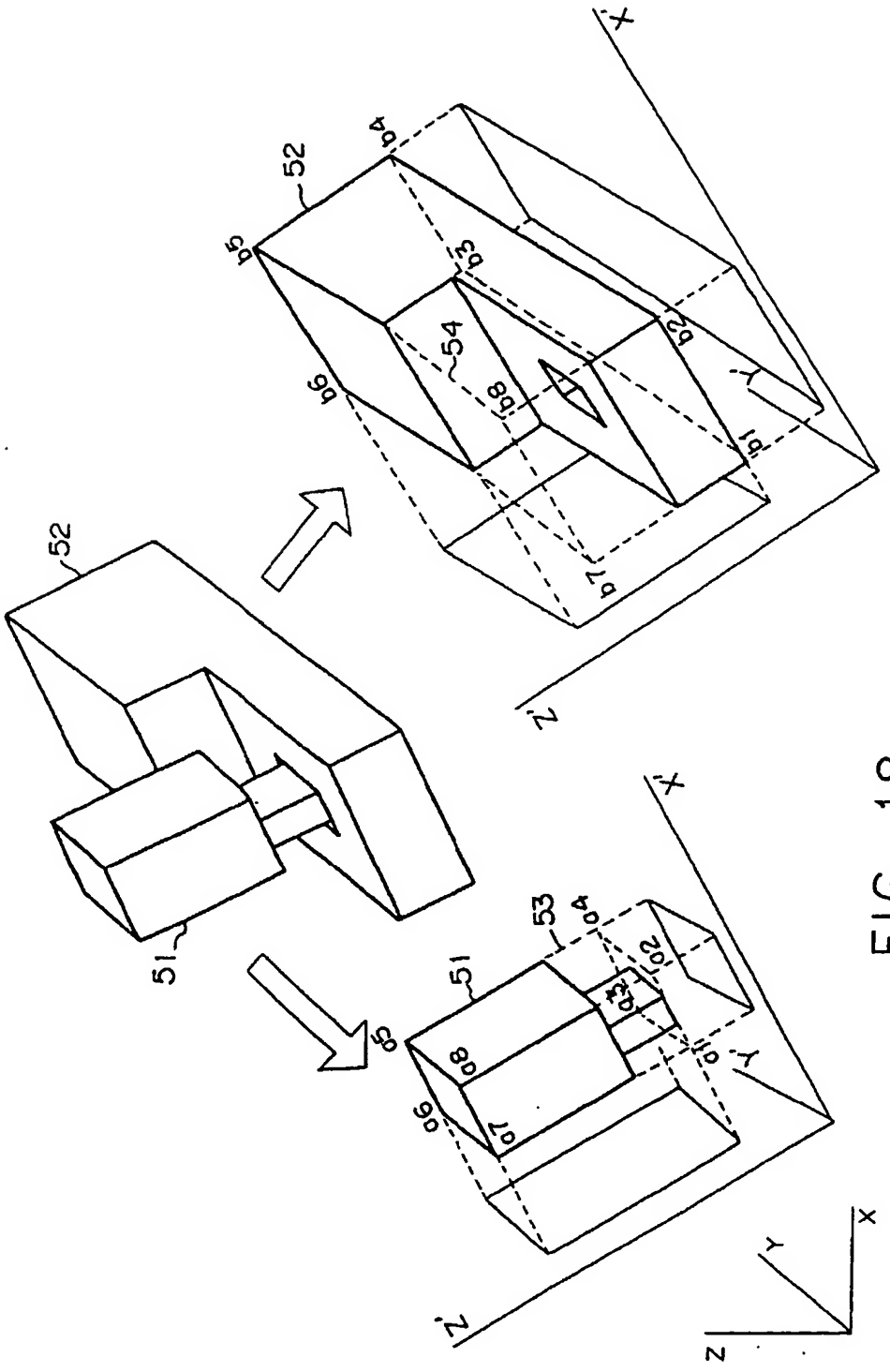

FIG. 18

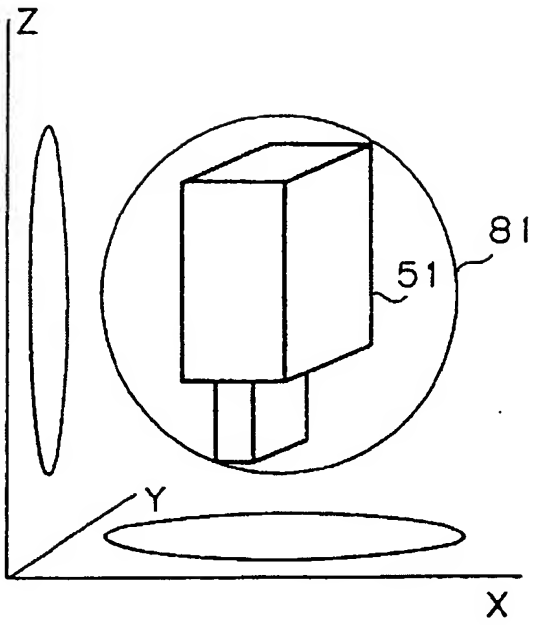

FIG. 19

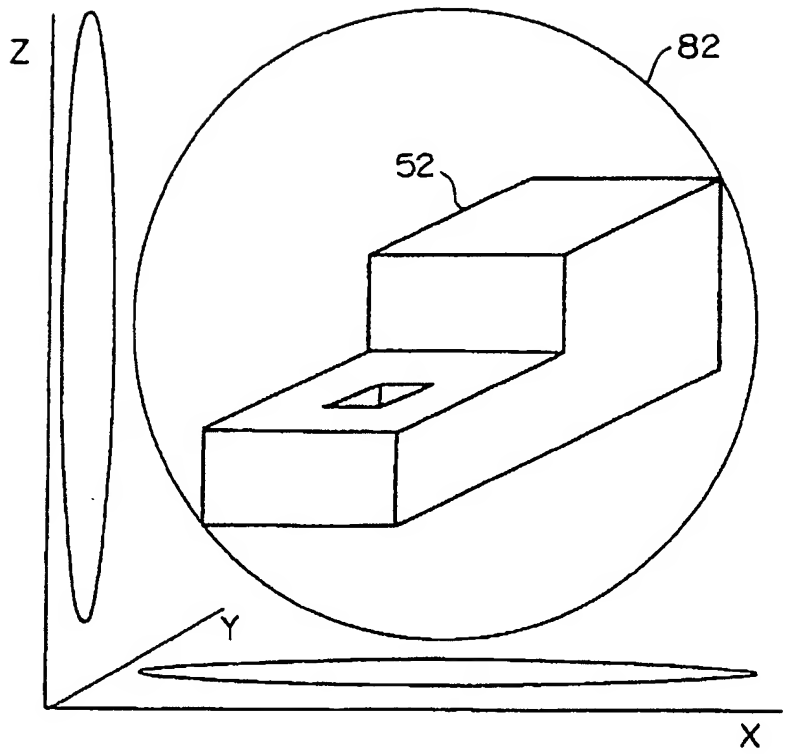

FIG. 20

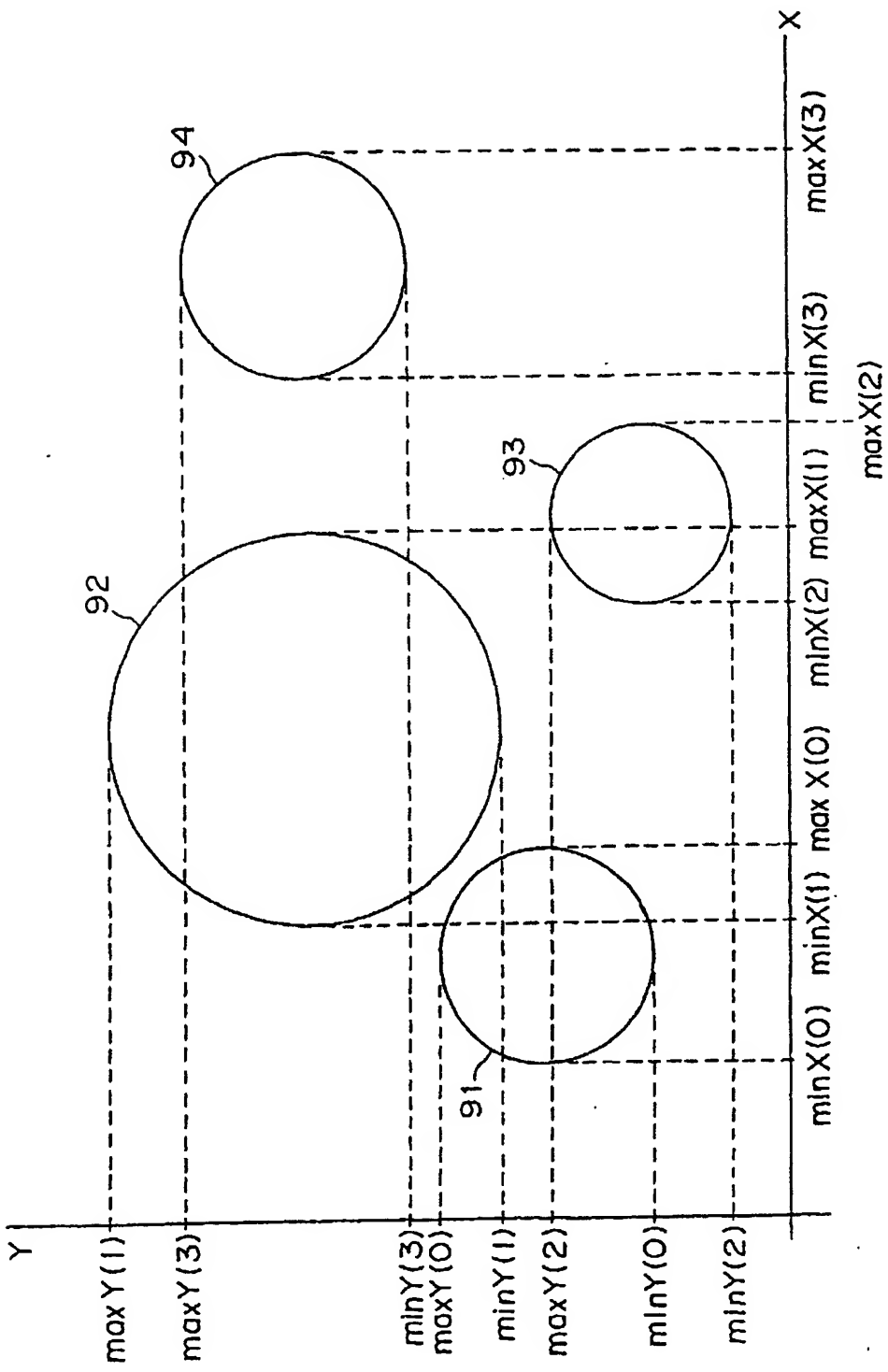
X
Y
94
92
93
91
maxX(3)
minX(3)
maxX(2)
maxX(1)
minX(2)
max X(0)
minX(1)
minX(0)
maxY(1)
maxY(3)
minY(3)
maxY(0)
minY(1)
maxY(2)
minY(0)
minY(2)

FIG. 21

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 957 452 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
01.10.2003 Bulletin 2003/40

(43) Date of publication A2:
17.11.1999 Bulletin 1999/46

(21) Application number: **99302255.7**

(22) Date of filing: **23.03.1999**

(51) Int Cl.[7]: **G06T 17/00**, G06T 17/40

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **12.05.1998 JP 12800798**

(71) Applicant: **FUJITSU LIMITED**
Kawasaki-shi, Kanagawa 211-8588 (JP)

(72) Inventor: **Sato, Yuichi, c/o Fujitsu Limited**
Kawasaki-shi, Kanagawa, 211-8588 (JP)

(74) Representative: **Fenlon, Christine Lesley et al**
Haseltine Lake & Co.,
Imperial House,
15-19 Kingsway
London WC2B 6UD (GB)

(54) Apparatus and method for checking interference of objects in computer

(57) An interference checking apparatus approximates the shape of an object with an envelope figure, such as a rectangular parallelepiped, a sphere, etc., and checks static interference among many objects at high speed. In this static interference check, an approximate figure is projected into a space of a lower dimension (2), pairs of objects with a possibility of interference are focused on in stages(6), and the number of the pairs is reduced. Then, a precise interference check is performed for the remaining pairs (4).

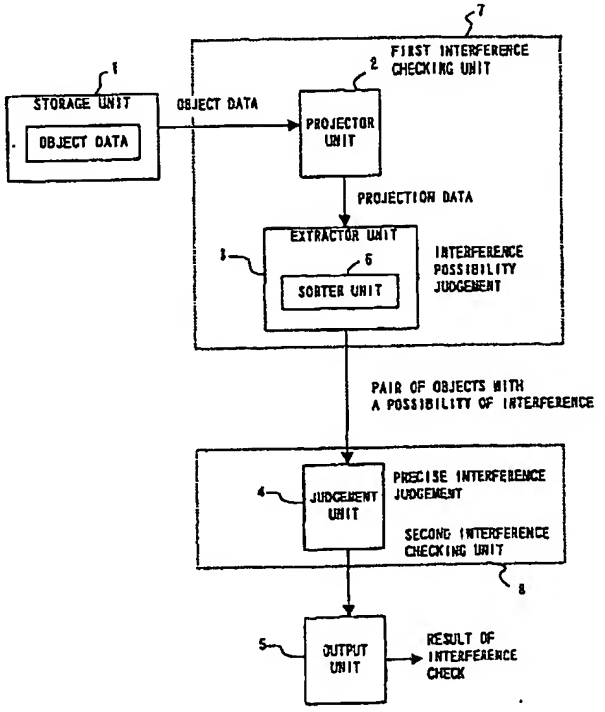

FIG. 1

EP 0 957 452 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 99 30 2255

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	COHEN J D ET AL: "I-COLLIDE: an interactive and exact collision detection system for large-scale environments" PROCEEDINGS 1995 SYMPOSIUM ON INTERACTIVE 3D GRAPHICS, PROCEEDINGS 1995 SYMPOSIUM ON INTERACTIVE 3D GRAPHICS, MONTEREY, CA, USA, 9-12 APRIL 1995, pages 189-196, 218, XP000546210 1995, New York, NY, USA, ACM, USA ISBN: 0-89791-736-7 * abstract; figure 2 *	1-7, 11-18	G06T17/00 G06T17/40
Y	* page 191 - page 192 *	8,9	
Y	* page 191, column 2, paragraph 2 *	10	
X	GARCIA-ALONSO A ET AL: "SOLVING THE COLLISION DETECTION PROBLEM" IEEE COMPUTER GRAPHICS AND APPLICATIONS, IEEE INC. NEW YORK, US, vol. 14, no. 3, 1 May 1994 (1994-05-01), pages 36-43, XP000511125 ISSN: 0272-1716 * page 37, column 1; figures 2,3 * * page 38 *	1,3-5, 11,12, 15-18	
Y	US 5 497 453 A (MEGAHED ABRAHAM E ET AL) 5 March 1996 (1996-03-05) * abstract; figures 6-8 *	8,9	
X,D A	US 5 056 031 A (NAKANO MASARU ET AL) 8 October 1991 (1991-10-08) * abstract *	11	
	-/--		

TECHNICAL FIELDS SEARCHED (Int.Cl.6)
G06T
B25J
G05B

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
MUNICH	23 July 2003	Meinl, W

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 99 30 2255

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
A	PARTHA SRINIVASAN ET AL: "COMPUTATIONAL GEOMETRIC METHODS IN VOLUMETRIC INTERSECTION FOR 3D RECONSTRUCTION" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION. SCOTTSDALE, MAY 15 - 19, 1989, WASHINGTON, IEEE COMP. SOC. PRESS, US, vol. 1, 15 May 1989 (1989-05-15), pages 190-195, XP000042721 ISBN: 0-8186-1938-4 * abstract * * page 191, column 1, line 1-20 *		
Y	PONAMGI M K ET AL: "Incremental algorithms for collision detection between polygonal models" IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, JAN.-MARCH 1997, IEEE, USA, vol. 3, no. 1, pages 51-64, XP000685845 ISSN: 1077-2626 * pp.60,61: section 4.8 * * abstract; figure 2 *	10	TECHNICAL FIELDS SEARCHED (Int.Cl.6)
Y	JP 09 225143 A (NAMCO LTD.) 2 September 1997 (1997-09-02) * abstract * & US 6 128 018 A 3 October 2000 (2000-10-03) * column 5, line 10-33; figures 7-11 *	10	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
MUNICH	23 July 2003	Meinl, W

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

Application Number
EP 99 30 2255

CLAIMS INCURRING FEES

The present European patent application comprised at the time of filing more than ten claims.

☐ Only part of the claims have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims and for those claims for which claims fees have been paid, namely claim(s):

☐ No claims fees have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims.

LACK OF UNITY OF INVENTION

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

see sheet B

☒ All further search fees have been paid within the fixed time limit. The present European search report has been drawn up for all claims.

☐ As all searchable claims could be searched without effort justifying an additional fee, the Search Division did not invite payment of any additional fee.

☐ Only part of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the inventions in respect of which search fees have been paid, namely claims:

☐ None of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the invention first mentioned in the claims, namely claims:

European Patent Office

LACK OF UNITY OF INVENTION
SHEET B

Application Number
EP 99 30 2255

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

1. Claims: 1-9, 11-18

   The claims relate to collision detection using a two-stage process.

2. Claim : 10

   The claim relates to dynamic collision detection using a special update procedure, without any mention of a two-stage process.

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 99 30 2255

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

23-07-2003

Patent document cited in search report		Publication date	Patent family member(s)		Publication date
US 5497453	A	05-03-1996	NONE		
US 5056031	A	08-10-1991	GB	2227106 A ,B	18-07-1990
			JP	2224004 A	06-09-1990
			JP	2826138 B2	18-11-1998
JP 9225143	A	02-09-1997	JP	3138423 B2	26-02-2001
			US	6128018 A	03-10-2000

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82